



Analyse et commande des systèmes paramétrés, par la fonction signe matricielle

Jérémie Guerra

► To cite this version:

Jérémie Guerra. Analyse et commande des systèmes paramétrés, par la fonction signe matricielle. Automatique. Ecole Centrale de Nantes (ECN), 2014. Français. NNT: . tel-01171266

HAL Id: tel-01171266

<https://hal.science/tel-01171266>

Submitted on 3 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Thèse de Doctorat

Jérémie GUERRA

*Mémoire présenté en vue de l'obtention
du grade de Docteur de l'Ecole Centrale de Nantes
Sous le label de l'Université Nantes Angers Le Mans*

École doctorale : Sciences et Technologies de l'Information et Mathématiques

Discipline : Automatique et Informatique Appliquée

Laboratoire : Institut de Recherche en Communications et Cybernétique de Nantes

Soutenue le 12 décembre 2014

Analyse et commande des systèmes paramétrés, par la fonction signe matricielle

JURY

Président :	Gilles DUC , Professeur à Supelec, Paris
Rapporteurs :	Gérard SCORLETTI , Professeur des universités, Ecole centrale de Lyon, Ampère Jamal DAAFOUZ , Professeur des universités, Université de Lorraine, CRAN
Examineurs:	Michel MALABRE , Directeur de Recherche au CNRS, Ecole Centrale de Nantes, IRCCyN Mohamed YAGOUBI , Maître Assistant à l'Ecole des Mines de Nantes
Directeur de thèse :	Philippe CHEVREL , Professeur de l'Ecole des Mines de Nantes

En mathématiques, on ne comprend pas les choses, on s'y habitue.

John von Neumann

*En essayant continuellement, on finit par réussir.
Donc plus ça rate, plus on a de chances que ça marche.*

Jacques Rouxel (Les Shadoks)

Remerciements

Tout d'abord, je tiens à remercier mes directeurs de thèse Philippe Chevrel et Mohamed Yagoubi pour leur soutien tout au long de ces trois années. Philippe m'a donné les clés nécessaires pour passer du statut d'étudiant professionnel au statut de chercheur débutant. Mohamed quant à lui, m'a donné les clés de la salle de TP... Plus sérieusement, Mohamed a su m'apporter un certain regard critique, et des idées aussi, ça c'est important, sans compter l'aide à la rédaction (sans lui, ce n'était pas gagné). Ils ont tous les deux fait preuve d'un encadrement de premier choix, ils ont toujours eu des réponses à m'apporter quand j'avais une idée en tête, et même quand j'en avais pas. Bref, ils m'ont permis de passer trois années sereines, tant sur le plan moral que scientifique. Je les remercie également de m'avoir laissé partir en conférences afin de découvrir le monde scientifique, et le monde tout court.

Je remercie mes deux rapporteurs, Gérard Scorletti et Jamal Daafouz pour leurs conseils suite à la lecture de la première version de ce manuscrit.

Je tiens à remercier également les secrétaires qui m'ont considérablement aidé dans les démarches administratives, Anita, Isabelle, Delphine, Evelyne...

Ensuite, mes deux collègues et amis Clément et Michaël pour le partage de la machine à café, que je remercie également, et leur soutien bien sûr, pour le partage d'une bière à la Cervoiserie, que je remercie également. Un grand merci aussi à tous les autres, Agnès, Axel, Carlos, Juliette, Majid, Mi, Tanguy, Renaud, Thomas, Yier (Dans l'ordre alphabétique pour qu'il n'y ait pas de jaloux)...

Enfin, le manuscrit n'aurait pas été ce qu'il est sans la présence à mes côtés de ma chère et tendre Angelica, qui est d'ailleurs très forte avec la numérotation des tableaux. Je remercie également mes parents, ma famille et mes amis.

Je remercie les (bons) écrivains, scientifiques, philosophes, romanciers, humoristes, qui je pense ont aussi fait de moi ce que je suis. Je terminerai ces remerciements en remerciant Douglas Adams, qui m'a permis de trouver la réponse à la grande question sur la vie, l'univers et le reste... 42.

Table des matières

INTRODUCTION	12
PANORAMA SUR LA FONCTION SIGNE MATRICIELLE	17
2.1 LES PREMIERS PAS	17
2.1.1 Définition	17
2.1.2 Propriétés	18
2.1.3 La notion de signe étendu	18
2.2 MÉTHODES POUR LE CALCUL DE LA FONCTION SIGNE MATRICIELLE	19
2.2.1 Les méthodes directes	19
2.2.1.1 Décomposition de Schur	19
2.2.1.2 Les formes intégrales	19
2.2.2 Les méthodes itératives	20
2.2.2.1 Les itérations de Newton	20
2.2.2.2 Les itérations de Newton-Schultz	21
2.2.2.3 Les itérations de la famille Padé	22
2.2.2.4 Une dérivation de l'écriture de Padé	25
2.2.2.5 <i>Scaling</i> sur les itérations de Newton	26
2.2.2.6 Les itérations dérivées des polynômes de Laurent	27
2.3 COMPARAISON DES MÉTHODES ITÉRATIVES AU CAS CONSTANT	30
2.4 FONCTION SIGNE MATRICIELLE POUR L'AUTOMATIQUE	33
2.4.1 Liens entre la fonction signe matricielle et la stabilité robuste	34
2.4.2 Liens entre la fonction signe matricielle et l'équation de Riccati	36
2.4.3 Liens entre la fonction signe matricielle et l'équation de Lyapunov	39
2.4.4 Liens entre la fonction signe matricielle et l'équation de Sylvester	42
2.4.5 Liens avec les LMI	43
2.5 CONCLUSION	44
MÉTHODES NON-ITÉRATIVES	45
3.1 MÉTHODE DIRECTE DE RÉOLUTION D'ÉQUATIONS DE LYAPUNOV	45
3.1.1 Définition du problème	46
3.1.1.1 Norme H_2 d'un système	46
3.1.1.2 Analyse H_2 robuste	47
3.1.1.3 Contrôle H_2 par retour d'état structuré	48
3.1.2 Résolution d'équations de Lyapunov paramétrées	48
3.1.2.1 Méthode d'inversion directe	48
3.1.2.2 Méthode de transformée de Fourier Discrète	52
3.1.2.3 Exemples numériques	54
3.2 FONCTION SIGNE MATRICIELLE PARAMÉTRÉE : DÉFINITION INTÉGRALE	58
3.2.1 Introduction	58
3.2.2 Définitions et méthodes, classe de matrices	59
3.2.3 Extension au cas paramétré	62
3.2.3.1 Fonction signe matricielle	62
3.2.3.2 Extension au cas multi-paramétrique	64
3.2.3.3 Stabilité robuste, Equations de Lyapunov et de Riccati	65

3.3	CAS PARTICULIER	68
3.3.1	<i>Ecriture de la fonction signe d'une matrice d'ordre 2</i>	68
3.3.2	<i>Résolution d'une équation de Lyapunov d'ordre 2</i>	69
3.4	FONCTION SIGNE ET SPECTRE MATRICIEL	70
3.4.1	<i>Introduction</i>	70
3.4.2	<i>Un algorithme pour la séparation de spectre</i>	71
3.4.2.1	Le cas de matrices diagonalisables à coefficients constants	71
3.4.2.1	Le cas de matrices déficientes à coefficients constants	75
3.4.3	<i>Une fonction signe matricielle pour la solution d'une équation de Sylvester paramétrique</i>	78
3.5	CONCLUSION	83
MÉTHODES ITÉRATIVES		84
4.1	APPROXIMATION DE LA FONCTION SIGNE MATRICIELLE PAR LES POLYNÔMES DE LAURENT	84
4.1.1	<i>Introduction</i>	84
4.1.2	<i>Préliminaires</i>	85
4.1.2.1	Rappel sur les itérations de Newton-Shultz (N-S)	85
4.1.2.2	Application aux équations de Riccati	86
4.1.3	<i>Problèmes ARE et CARE constants</i>	87
4.1.4	<i>Un approximant rationnel de la fonction signe matricielle</i>	88
4.1.5	<i>Méthode particulière pour le calcul de racines matricielles</i>	93
4.1.6	<i>Extension au cas paramétré</i>	95
4.2	COMBINAISON DE LA FONCTION SIGNE MATRICIELLE ET DE L'INVERSION DFT	100
4.2.1	<i>Introduction</i>	100
4.2.2	<i>Classe des matrices</i>	100
4.2.3	<i>Inversion de matrices polynomiales par la méthode de Transformation de Fourier Discrète</i>	102
4.2.4	<i>L'utilisation de la fonction signe matricielle</i>	105
4.2.5	<i>Un filtre de Kalman paramétrique estimant les harmoniques d'amplitude d'ondes</i>	109
4.2.6	<i>Modèle paramétré d'une aile flottante</i>	112
4.3	LA FONCTION SIGNE MATRICIELLE ÉTENDUE POUR L'ÉTUDE DU DOMAINE DE STABILITÉ	115
4.3.1	<i>Introduction</i>	115
4.3.2	<i>La fonction signe étendue</i>	115
4.3.3	<i>La fonction signe matricielle paramétrique</i>	117
4.3.3.1	Inverse de Drazin généralisée	117
4.3.3.2	Fonction signe matricielle paramétrique étendue	118
4.3.4	<i>Un algorithme pour la matrice signe paramétrique</i>	119
4.4	CONCLUSION	122
CONCLUSION		123
BIBLIOGRAPHIE		126

Notations, symboles et abréviations

\mathbb{C}	ensemble des nombres complexes
\mathbb{R}	ensemble des nombres réels
\mathbb{R}^*	ensemble des nombres réels non nuls
$\mathbb{R}^{n \times m}$	ensemble des matrices réelles de taille $n \times m$
$\mathbb{C}^{n \times m}$	ensemble des matrices complexes de taille $n \times m$
\in	‘appartient à’
\otimes	produit de Kronecker
$sign(\cdot)$	signe d’une matrice ou d’un scalaire
$exsign(\cdot)$	signe étendu d’une matrice ou d’un scalaire
$gsign(\cdot)$	signe généralisé d’une matrice ou d’un scalaire
$X(\theta)$	matrice X dépendante de θ
I_n	matrice identité de taille $n \times n$
$0_{n \times m}$	matrice zéro de taille $n \times m$
X^T	transposée de la matrice X
$\det(\cdot)$	déterminant d’une matrice
$\text{Re}(\cdot)$	partie réelle d’un nombre complexe
$diag(\cdot)$	matrice diagonale par bloc
$trace(\cdot)$	trace d’une matrice
$\lambda \left(\lambda^* \right)$	valeurs propres (valeurs propres non nulles) d’une matrice
$\ \cdot\ $	norme d’une matrice
\tanh	tangente hyperbolique
$\rho(\cdot)$	rayon spectral d’une matrice
$L(z)$	polynôme de Laurent en z
$Tr(\cdot)$	trace d’une matrice
\triangleq	égal par définition

$\sigma(\cdot)$	spectre d'une matrice
\emptyset	ensemble vide
$\left[\begin{array}{c c} A & B \\ \hline C & D \end{array} \right]$	matrice système
N-S	Newton-Shultz
LTI	Linéaire à Temps Invariant (de l'anglais : Linear Time-Invariant)
DFT	Transformée de Fourier Discrète (de l'anglais : Discrete Fourier Transform)
LMI	Inégalités Linéaires Matricielles (de l'anglais : Linear Matrix Inequalities)
LFT	Transformation Fractionnelles Linéaires (de l'anglais : Linear Fractional Transform)
ARE	Equation de Riccati Algébrique (de l'anglais : Algebraic Riccati Equation)
CARE	Equation de Riccati Algébrique Contrainte (de l'anglais : Algebraic Riccati Equation constraint)
PDRE	Equation de Riccati Paramétrée (de l'anglais Parameter-Dependent Riccati Equation)
SDRE	Equation de Riccati dépendant de l'état (de l'anglais State-Dependent Riccati Equation)
SOS	Somme de carrés (de l'anglais Sum of Squares)

Liste des figures

Fig. 2.1 : la fonction $y = \tanh(k \operatorname{arctanh}(x))$ pour différentes valeurs de k	23
Fig. 2.2 : La spirale \tanh en rouge et la séquence de Padé en bleu pour $k=2$ et $z_0 = 0.2 + 0.8i$	24
Fig. 2.3: Temps de calcul en fonction de la taille des matrices.....	31
Fig. 2.4: Nombre d'itérations en fonction de la taille des matrices.....	31
Fig. 3.1: Performance H_2 robuste.	56
Fig. 4.1: Points scalaires vérifiant la condition (4.10).....	91
Fig. 4.2: La vérification de la condition de N-S pour la première approximation rationnelle de la fonction signe.....	96
Fig. 4.3: L'erreur $\ \tilde{S}(\theta_i) - S_N(\theta_i)\ _2$	97
Fig. 4.4: La vérification de la condition N-S pour la première approximation rationnelle de la racine matricielle.	98
Fig. 4.5: L'erreur $\ \tilde{S}^{1/2}(\theta_i) - S_N^{1/2}(\theta_i)\ _2, \theta_i \in [-1.5, 1.5]$	99
Fig. 4.6: L'erreur en fonction des paramètres θ_1, θ_2	108
Fig. 4.7: L'erreur en fonction du paramètre w	111
Fig. 4.8: Evolution de la partie réelle des valeurs propres de la solution $X(w)$	112
Fig. 4.9: Evolution des valeurs propres en boucle ouverte et fermée.	114
Fig. 4.10: Région de Stabilité pour $Z_1(\theta)$	120
Fig. 4.11: Région de Stabilité pour $Z_2(\theta)$	121
Fig. 4.12: Région de stabilité (en bleu) utilisant l' Algorithme 4.4 pour NN5	122
Fig. 4.13: Région de stabilité (en bleu) utilisant l' Algorithme 4.4 pour AC7	122

Liste des tables

Table 2.1 : Fonctions de Padé en regard de l et m	24
Table 2.2 : Le coefficient \bar{F}_0 de $L(z;S,T)^{-1}$	30
Table 2.3 : Comparaison des différentes méthodes de calcul de la fonction signe matricielle	32
Table 2.4 : Comparaison du temps de calcul entre LMI et la fonction signe matricielle.....	43
Table 4.1 : Résultats pour l'Exemple 4.1.	92
Table 4.2 : Résultats pour l'Exemple 4.2.	92

Publications personnelles

Articles de conférences avec actes et comité de lecture

J. Guerra, M. Yagoubi and P. Chevrel. A matrix sign function framework for robust stability analysis and parameter-dependent Lyapunov and Riccati equalities. *IEEE Conference on Decision and Control, Maui, Hawaii*, pp. 3490-3495, 2012.

J. Guerra, M. Yagoubi and P. Chevrel. Parametric Lyapunov equation approach for robust H_2 analysis and structured H_2 control problems. *IFAC System Structure and Control, Grenoble*, vol. 5, pp. 665-670. 2013.

J. Guerra, M. Yagoubi and P. Chevrel. A matrix sign function based solution of parameter dependent Sylvester equations. *European Control Conference, Strasbourg* pp. 400-405, 2014.

J. Guerra, M. Yagoubi and P. Chevrel. Toward a rational matrix approximation of the parameter-dependent Riccati equation solution. *19th IFAC World Congress, Cape Town, South Africa*, 2014.

Articles de conférences nationales

J. Guerra, M. Yagoubi and P. Chevrel. Commande robuste et résiliente. *GDR JDJN MACS Strasbourg*. 2013.

J. Guerra, M. Yagoubi and P. Chevrel. Résolution d'équations de Lyapunov et de Riccati paramétrées : une approche par la fonction signe matricielle. *JDOC Saint Nazaire*. 2013.

Articles soumis en journal

J. Guerra, M. Yagoubi and P. Chevrel. Mixed Matrix Sign Function/DFT Inversion Method for Solving Parameter-dependent Riccati Equation. *Journal of the Franklin Institute, Special Issue on Matrix Equations*, 2014.

J. Guerra, M. Yagoubi and P. Chevrel. Toward a Rational Matrix Approximation of the Parameter-Dependent Riccati Equation Solution. *International Journal of Systems Science*, 2014.

Chapitre 1 :

Introduction

*Chaque chose au monde porte en elle sa réponse,
ce qui prend du temps ce sont les questions.*

José Saramago

Après deux décennies où l'optimisation convexe sous contraintes LMI [BE-GF+94], grâce aux algorithmes de résolution efficaces sous-jacents, a représenté un outil *presque* inévitable pour l'analyse et/ou la commande des systèmes LTI à travers une approche unifiée et générique, nous avons assisté à un élargissement des classes de systèmes traitées par cet outil [Sco97]. Des travaux, tels que ceux de la thèse de Marc Dinh [Din05] à titre d'exemple, ont apporté une pierre à l'édifice en permettant la mise sous forme de problèmes d'optimisation convexe sous contraintes LMI indépendant du vecteur des paramètres pour tous les problèmes d'analyse et de commande se formulant comme des problèmes d'optimisation convexe sous contraintes LMI paramétrées (les paramètres sont supposés appartenir à un ensemble donné et être constants dans le temps). Depuis, l'analyse et la commande, à titre d'exemple, des systèmes LTV, de certains systèmes non linéaires et/ou de dimension infinie sont devenues envisageables sous ce formalisme LMI. La classe des systèmes Linéaires à Paramètres Variant, une sous-classe importante des systèmes LTV, a fait l'objet d'un très grand nombre de résultats sous ce même formalisme (voire, sans prétendre l'exhaustivité, les références incluses dans la thèse d'Anis Bouali [Bou08]). Plus récemment les travaux de [Che11], à titre d'exemple, ont élargi le spectre des problèmes considérés en utilisant la programmation SOS (de l'anglais Sum of Squares) i.e. une technique d'optimisation qui a été développée initialement par [Par00] et qui se base sur l'idée *simple* qu'un polynôme est non négatif du moment qu'on peut le représenter sous forme d'une somme de carrés de polynômes. Un moyen donc permettant de trouver des fonctions de Lyapunov (polynomiales ou à dépendance paramétrique plus *complexes*) prouvant la stabilité du système considéré. Il a été ainsi possible de se ramener au formalisme LMI dans le cadre des systèmes polynomiaux, des systèmes polynomiaux incertains et même certains systèmes non-linéaires non polynomiaux (incertains).

Le point fort de ce formalisme réside donc dans sa généralité et dans le fait qu'il inclut la notion de complexité. Ainsi, un problème d'analyse ou de commande peut être résolu efficacement dès lors qu'il est équivalent à un problème d'optimisation convexe sous contraintes LMI. Partant de ce constat, il a été *implicitement* admis que le formalisme LMI surclasse les formulations utilisant par exemple des équations de Lyapunov/Sylvester ou de

Riccati puisqu'il offre des possibilités *évidentes* de formulation d'un grand nombre de problèmes d'analyse et de commande dépassant le cadre strict des systèmes LTI.

Cependant, plusieurs verrous méthodologiques, théoriques et numériques semblent militer pour la recherche d'une autre approche peut être moins générique mais plus avantageuse pour l'analyse et la commande des systèmes paramétrés ou du moins de certaines classes de systèmes paramétrés. Sans être exhaustifs, nous citons dans ce qui suit certains de ces verrous : (i) Certains problèmes de commande nécessitent le réglage du régulateur pendant l'exploitation de ce dernier sans autoriser la possibilité d'une nouvelle synthèse en ligne. Nous pensons ici à la synthèse de régulateurs (multivariables) réglables sujets à des cahiers de charges complexes pour lesquels des méthodes de synthèse avancées du type H_2 ou H_∞ sont nécessaires. (ii) La résolution *en ligne* des problèmes d'optimisation convexes sous contraintes LMI est *rarement* envisageable, malgré les avancées (logicielles et matérielles) enregistrées au niveau informatique pour le traitement par les calculateurs d'un grand nombre d'opérations à la seconde. (iii) Le paradigme de Lyapunov, la brique fondamentale liant le formalisme LMI aux problèmes d'analyse et de commande des systèmes introduit intrinsèquement des variables de décision supplémentaires alors qu'il peut parfois, dans le cas de la stabilité et la stabilisation, être évité. Effectivement, pour la stabilité voire la stabilisation, on peut directement analyser le signe matriciel de la matrice d'état sans faire appel à la matrice de Lyapunov. (iv) Les outils alternatifs tels que les équations de Lyapunov ou de Riccati peuvent parfois être résolus plus efficacement et donc peuvent être avantageusement utilisés.

Pour ces raisons nous nous sommes permis dans ce travail de thèse d'ouvrir la voie vers une approche privilégiant les formulations par équations de Lyapunov/Sylvester et de Riccati étendues aux cas paramétrés. Par ailleurs, pour étayer encore plus les propos ci-dessus nous souhaitons passer en revue, sans les détailler, certains problèmes *particuliers* d'analyse et de commande des systèmes faisant appel *intrinsèquement* à la résolution d'une équation de Lyapunov/Sylvester ou de Riccati dépendant de paramètres. Nous notons, toutefois, que ces cas ne seront pas forcément traités dans le cadre de ce travail de thèse mais constituent des cas ayant suscité l'intérêt de l'étude objet de cette thèse. Un premier cas qui nous semble pertinent concerne la stratégie de commande nécessitant la résolution d'une équation de Riccati dépendant de l'état (SDRE) de l'anglais State-Dependent Riccati Equation. Un état de l'art assez exhaustif sur cette stratégie et ses applications est proposé dans [Çim08]. La stratégie de commande via SDRE permet d'avoir une approche effective pour la synthèse de lois de commande non linéaires tout en permettant des non linéarités sur les états du système et une grande flexibilité due à l'utilisation de matrices de pondérations elles-mêmes dépendantes de l'état. A l'origine cette méthode a été proposée par [Pea62] et a depuis connu un très grand essor. Il s'agit d'une commande optimale non linéaire à horizon infini où le système est totalement observable, non linéaire en l'état et affine en l'entrée. La linéarisation étendue proposée par [Fri96], permet de factoriser le système non linéaire ainsi décrit sous forme d'un système linéaire avec des matrices d'état dépendantes de l'état. Plusieurs algorithmes de résolution en ligne de ces SDRE ont été

proposés dans la littérature (voire à titre d'exemple [IST+04]) mais le sujet reste ouvert et d'actualité. Le deuxième cas concerne les systèmes linéaires bi-dimensionnels 2D (ou multi-dimensionnels) qui ont été introduits pour la première fois dans les années 70 par [FM76] et [GR72]. Ils ont été utilisés en traitement d'image, filtrage des données [Roe75] et même en modélisation des équations aux dérivées partielles [Mar84], etc. Les conditions nécessaires et suffisantes de stabilité proposées dans [Aga90] peuvent être intuitivement reformulées sous formes d'équations de Lyapunov dépendantes des paramètres (notons ici que le paramètre peut être la variable de Laplace).

Ainsi, nous envisageons deux démarches pour la résolution d'équations de Lyapunov/Sylvester ou de Riccati associées à des problèmes d'analyse ou de commande de systèmes paramétrés. Une approche dite « directe » se basant sur des méthodes d'inversion directe de matrices dépendantes de paramètres [JL82]. Une approche dite « indirecte » se basant sur le calcul de fonction signe matricielle dans le cas paramétré. Notant ici qu'une première tentative d'utilisation de la fonction signe matricielle, dans le cas mono-paramétrique, a eu lieu dans les travaux de thèse J. K. Rice [RV10].

La fonction signe matricielle étant la clé de cette thèse, il est nécessaire ici d'en donner un court état de l'art. L'utilisation première de la fonction signe matricielle est de trouver les sous-espaces invariants positifs et négatifs d'une matrice. Un grand nombre de problèmes en automatique et dans la décomposition en valeurs propres / vecteurs propres ont des solutions qui peuvent être données en fonction de ces sous-espaces. Le premier vrai pas a été fait dans les travaux de Zolotarjov [Zol77] en 1877, où il caractérisait la meilleure approximation rationnelle de la fonction signe scalaire en terme de fonctions de sinus. Le deuxième pas, plus grand celui-ci, a été fait dans les travaux de Roberts en 1971, au niveau de l'introduction de la fonction signe matricielle, dans le domaine de l'automatique. Néanmoins, ces travaux n'ont été publiés qu'en 1980 [Rob80]. Se basant sur les travaux de Roberts, la recherche sur la fonction signe matricielle a évolué dans plusieurs domaines (Théorie de l'approximation et méthodes de calcul, estimation, applications à la commande, décomposition de valeurs propres, racines de matrices...). Pour la partie qui nous intéresse le plus dans cette thèse, à savoir les applications à la commande, nous noterons les travaux de Denman et Beavers [BD74], [BD76], [DB76], [Den76], dans la continuité des travaux de Roberts. Hoskins *et al.* [HMW77a], [HMW77b] ont utilisé la fonction signe comme une base pour un algorithme résolvant les équations de Sylvester et de Lyapunov. Finalement on notera les travaux de Gardiner et Laub [Gar88], [GL91], [LG88] qui utilisèrent les itérations de Newton pour implémenter la fonction signe matricielle dans un algorithme de résolution d'équations de Riccati algébriques (ARE).

Le manuscrit s'organisera de la manière suivante. Tout au long de la thèse, des méthodes ont été développées afin de résoudre des équations de Lyapunov, de Riccati et de Sylvester étendues au cas paramétré. L'objectif principal a été de s'intéresser à ces résolutions d'équations à l'aide principalement de la fonction signe matricielle. C'est pourquoi dans un premier temps, il a été nécessaire de faire un rappel sur les définitions et

les concepts de base liés à cette méthode, que l'on retrouvera dans le chapitre 2. Les méthodes développées se classent dans plusieurs catégories. Nous avons les méthodes dites d'« inversion directe », de calcul de la fonction signe intégrale, une méthode de « séparation de spectre » et enfin les méthodes de calcul de la fonction signe itératives.

L'unique méthode ici ne se basant pas sur la fonction signe matricielle est la méthode dite d'« inversion directe ». Celle-ci se base sur une écriture vectorielle de l'équation de Lyapunov, le but étant de se ramener à un simple problème d'inversion de matrice. La méthode de « séparation de spectre » utilise la fonction signe matricielle de manière indirecte. Cette partie est dédiée à la résolution d'équations de Sylvester. Cette méthode repose sur l'extraction de valeurs propres d'une matrice (que l'on peut faire à l'aide de la fonction signe). Les méthodes intégrales sont les calculs de la fonction signe matricielle « exactes ». Finalement, les méthodes itératives sont donc des manières approchées du calcul de la fonction signe. Plusieurs types d'itérations ont été repris/développés dans le chapitre 2.

Le manuscrit a donc été ordonné de la manière suivante. Dans un premier temps, l'étude de la fonction signe matricielle comprenant ses définitions et propriétés, puis un large panel, probablement non exhaustif, des méthodes de calcul de la fonction signe matricielle existantes, au cas constant tout d'abord. Les méthodes sont accompagnées d'exemples didactiques aidant à la compréhension de chacune d'entre elles. Finalement, la fin de ce chapitre introduit les relations existantes entre les équations de Lyapunov, de Riccati et de Sylvester avec la fonction signe matricielle. Les chapitres suivants reprendront toutes les méthodes décrites précédemment afin de résoudre des équations paramétrées. Chaque partie se concentrera sur un type d'équation en particulier.

Le chapitre 3 reprendra les méthodes non-itératives. C'est-à-dire, dans un premier temps sera décrit la méthode dite d'« inversion directe », que l'on étendra au cas paramétrique, afin de résoudre des équations de Lyapunov et de Riccati particulière pour l'analyse H_2 robuste et également du contrôle H_2 par retour d'état structuré. La deuxième partie de ce chapitre utilisera l'écriture intégrale de la fonction signe matricielle pour l'analyse robuste, la résolution d'équations de Lyapunov et de Riccati paramétrées, la troisième partie est une manière directe pour trouver la fonction signe matricielle d'une matrice d'ordre 2, dérivée de l'écriture intégrale. La dernière partie du chapitre 3 sera dédié à la séparation du spectre à l'aide de la fonction signe matricielle pour le calcul d'équations de Sylvester paramétrées particulières. Enfin, le chapitre 4 comprendra les différentes méthodes itératives développées. La première partie se consacrera à l'emploi des polynômes de Laurent combinés avec les itérations de Newton-Shultz. Cette partie vise à résoudre des équations de Riccati de deux types, standard (ARE) et particulière (sous contrainte) notée (CARE). Le traitement du calcul d'une racine matricielle y est également exposé. La deuxième partie se base sur une combinaison entre la fonction signe matricielle et les inverses de Fourier (DFT) pour la solution d'équations de Riccati paramétrées.

Finalement, une représentation matricielle du signe étendu pour étudier des régions de stabilité sera présentée dans la dernière partie de ce chapitre.

Après cette introduction descriptive des contenus des différents chapitres, nous proposons en quelques lignes un résumé des contributions de ce travail de thèse par rapport à l'existant. Si l'utilisation de la fonction signe matricielle pour la résolution d'équations de Riccati/Lyapunov existaient dans le cas constant, sa généralisation au cas paramétré était balbutiante dans la littérature. Nous avons, à travers ce travail de thèse, proposé différentes approches allant de la résolution exacte via la définition intégrale de la fonction signe matricielle jusqu'à des méthodes hybrides numériques/symboliques. Les différentes approches proposées ont en commun la possibilité de procéder par des opérations sur des matrices ne dépendant pas des paramètres. L'originalité dans un premier temps concernait l'utilisation de la définition intégrale combinée avec le théorème de Cayley-Hamilton pour le calcul d'inverses de matrices. Dans un second temps, une démarche originale pour la résolution d'équations de Sylvester non-standard a été proposée par le biais de la séparation de spectre à l'aide de la fonction signe matricielle. Dans le but de minimiser le nombre d'inversions matricielles, un approximant rationnel de la fonction signe matricielle paramétrée a été proposé pour la première fois via une approche par polynômes de Laurent. Des applications dans le cas de résolution d'équations de Riccati avec et sans contraintes ont été également proposées. Par ailleurs, une autre méthode itérative a été considérée incluant pour la première fois des techniques connues d'inversion de matrices par DFT et des itérations basées sur des approximations de Padé. Tout au long de ce travail de thèse, nous avons tenté de dépasser les limites intrinsèques de chacune des méthodes proposées. Les principaux verrous étant la complexité numérique, la dépendance paramétrique, le champ d'application, etc. Nous espérons que ce travail ouvre la voie vers une vraie alternative aux méthodes basées sur les contraintes LMI.

Chapitre 2:

Panorama sur la fonction signe matricielle

*Ce n'est pas dans la science qu'est le bonheur,
mais dans l'acquisition de la science.*

Edgar Allan Poe

Cette partie présente l'état de l'art de la fonction signe matricielle. Elle a pour but de faire un rappel général sur ses définitions et propriétés diverses. Différentes méthodes de calcul de la fonction signe matricielle sont répertoriées ainsi que des exemples permettant au lecteur de faire le point sur chacune d'entre elles. La dernière partie se concentrera sur les liens existants entre l'automatique et la fonction signe matricielle, principalement les équations de Lyapunov, de Riccati et de Sylvester. On considérera tout au long de ce chapitre des matrices ou scalaires ne dépendant pas des paramètres.

2.1 Les premiers pas

2.1.1 Définition

Originellement introduite par [Rob80], la fonction signe est définie pour $z \in \mathbb{C}^*$ par :

$$\text{sign}(z) = \begin{cases} 1, & \text{Re } z > 0 \\ -1, & \text{Re } z < 0 \end{cases} \quad (3.1)$$

Il est important de noter que si la partie réelle de z est nulle, son signe n'est pas défini. Considérons maintenant une matrice $A \in \mathbb{C}^{n \times n}$ quelconque sans valeurs propres à partie réelle nulle. On peut ensuite définir une forme canonique de Jordan de la forme $A = ZJZ^{-1}$ avec $J = \text{diag}(J_1, J_2)$ où J_1 et J_2 sont des matrices carrées avec des valeurs propres respectivement dans \mathbb{C}^- et \mathbb{C}^+ . Le signe de la matrice A est ensuite défini par :

$$\text{sign}(A) = Z \begin{bmatrix} -I_{J_1} & 0 \\ 0 & I_{J_2} \end{bmatrix} Z^{-1} \quad (3.2)$$

avec I_{J_1} et I_{J_2} les matrices identité de dimensions égales à J_1 et J_2 respectivement.

2.1.2 Propriétés

En notant $S = \text{sign}(A)$, et en supposant que $A \in \mathbb{C}^{n \times n}$ n'a pas de valeurs propres sur l'axe des imaginaires, les propriétés suivantes sont vraies :

- (a) $S^2 = I_n$, (S est involutif).
- (b) S est diagonalisable avec des valeurs propres égales à ± 1 .
- (c) S et A commutent : $SA = AS$.
- (d) Si A est réel, S est réel.
- (e) Si c est un scalaire différent de 0, alors $\text{sign}(cA) = \text{sign}(c) \cdot \text{sign}(A)$.
- (f) $(I_n + S)/2$ et $(I_n - S)/2$ sont des projecteurs dans les sous-espaces invariants associés, avec respectivement des valeurs propres dans le demi plan droit et le demi plan gauche.
- (g) Si V est une matrice non singulière, alors $\text{sign}(V^{-1}AV) = V^{-1}\text{sign}(A)V$.
- (h) $\text{sign}(A^T) = (\text{sign}(A))^T$.

2.1.3 La notion de signe étendu

De manière générale, dans la littérature, la définition de la fonction signe se présente sous la forme (3.1), c'est-à-dire, en ne définissant pas le signe d'une matrice contenant au moins une valeur propre à partie réelle nulle. Néanmoins, une définition nous permet de contourner ce problème et d'intégrer également des matrices contenant des valeurs propres à partie réelle nulle. Cette définition a été premièrement décrite par [Mat78]. Le signe est décrit de la manière suivante :

$$\text{exsign}(z) = \begin{cases} 1, & \text{si } \text{Re}(z) > 0 \\ 0, & \text{si } \text{Re}(z) = 0 \\ -1, & \text{si } \text{Re}(z) < 0 \end{cases} \quad (3.3)$$

La fonction exsign est discontinue et il est donc impossible d'utiliser la décomposition en blocs de Jordan. Afin d'étendre cette définition au cas matriciel, il est nécessaire de définir un scalaire ε suffisamment petit tel que $\text{sign}(A \pm \varepsilon I_n)$ soient tous les deux définis. exsign est donc donné par :

$$\text{exsign}(A) = \frac{1}{2} (\text{sign}(A + \varepsilon I) + \text{sign}(A - \varepsilon I)) \quad (3.4)$$

Le choix du scalaire ε est crucial pour la convergence des itérations. En effet, plus ε est petit, plus les inverses de $(A \pm \varepsilon I_n)$ ont des chances d'être mal conditionnées. On pourra prendre en compte par exemple un choix de ε défini dans [KL95] par :

$$\varepsilon < \min |\lambda^*| \quad (3.5)$$

avec $\lambda^* \neq 0$ correspondant aux valeurs propres non nulles de A . D'autres choix possibles de ε sont décrits dans [How83], [Sti91].

Une fois le scalaire ε défini, le signe étendu de A est obtenu directement par la somme de deux fonctions signe. Dans les méthodes de calcul décrites dans la prochaine section, les itérations sont présentées suivant la définition (3.1), chacune de ces méthodes est bien entendu généralisable pour le cas de la fonction signe étendue.

Après ces quelques définitions, plusieurs méthodes de calcul de la fonction signe matricielle sont décrites dans la section suivante.

2.2 Méthodes pour le calcul de la fonction signe matricielle

Dans cette partie on fera la distinction entre les méthodes dites « directes », c'est-à-dire n'utilisant ni approximation ni itérations et les méthodes itératives. Pour le cas des méthodes itératives, chacune d'entre elle sera accompagnée d'un exemple.

2.2.1 Les méthodes directes

2.2.1.1 Décomposition de Schur

La première méthode proposée pour le calcul direct de la fonction signe, (en dehors de la décomposition en blocs de Jordan) est la décomposition de Schur. Considérons

$$A = U \begin{bmatrix} N & T \\ 0 & P \end{bmatrix} U^T \quad (3.6)$$

où U est une matrice orthogonale, N et P ont respectivement des valeurs propres dans \mathbb{C}^- et \mathbb{C}^+ . Le signe de A est alors donné par :

$$\text{sign}(A) = U \begin{bmatrix} -I_N & S \\ 0 & I_P \end{bmatrix} U^T \quad (3.7)$$

où S est solution de l'équation de Sylvester suivante :

$$NS - SP = -2T \quad (3.8)$$

2.2.1.2 Les formes intégrales

Les formes présentées ici sont tirées de [Rob80] et [KL95]. La première forme est définie par :

$$\text{sign}(A) = \frac{2}{\pi} A \int_0^{+\infty} (y^2 I_n + A^2)^{-1} dy \quad (3.9)$$

La seconde est dérivée de la première en faisant le changement de variable $y = \tan \delta$:

$$\text{sign}(A) = \frac{2}{\pi} A \int_0^{\pi/2} (\sin(\delta)^2 I_n + \cos(\delta)^2 A^2)^{-1} d\delta \quad (3.10)$$

2.2.2 Les méthodes itératives

2.2.2.1 Les itérations de Newton

Cette méthode est la plus utilisée des méthodes itératives pour le calcul de la fonction signe. Elle se définit par l'algorithme suivant :

$$\begin{cases} X_{k+1} = \frac{1}{2} (X_k + X_k^{-1}), & X_0 = A \\ \lim_{k \rightarrow \infty} (X_k) = \text{sign}(A) \end{cases} \quad (3.11)$$

La convergence peut être lente si A a des valeurs propres proches de l'axe imaginaire. Pour prendre en compte ce phénomène, des méthodes spécifiques ont été développées comme par exemple les itérations de Newton-Schulz, les itérations de Padé ainsi que plusieurs versions pondérées de la méthode de Newton. Etant donné que la définition ici des itérations de Newton implique des itérations infinies, nous considérerons par la suite un critère d'arrêt afin de stopper les itérations. Nous avons choisi le suivant, repris des travaux de Higham [Hig08]:

$$\delta_{k+1} := \frac{\|X_{k+1} - X_k\|}{\|X_{k+1}\|} \leq \varepsilon \quad (3.12)$$

Où ε correspond au degré de précision que l'on veut atteindre.

Exemple 2.1:

Considérons les deux matrices suivantes :

$$A_1 = \begin{bmatrix} 1 & 2 & 4 & -2 \\ 6 & 3 & -4 & 2 \\ -2 & 1 & 8 & 3 \\ 1 & 0 & 7 & -5 \end{bmatrix} \quad \text{et} \quad A_2 = \begin{bmatrix} 1 & 2 & 3 & -2 \\ 0 & 3 & -4 & 2 \\ 1 & 1 & 8 & 3 \\ 2 & 0 & 7 & 5 \end{bmatrix}$$

Dont on désire calculer le signe matriciel. Les itérations de Newton définies par (3.11) sont appliquées pour chacune d'elles. Les itérations sont stoppées pour une précision de $\varepsilon = 10^{-8}$. Cela donne leurs signes respectifs:

$$\text{sign}(A_1) = \begin{bmatrix} -0.398 & 0.570 & 0.513 & 0.091 \\ 1.837 & 0.255 & -0.599 & -0.258 \\ -0.374 & 0.140 & 0.887 & 0.489 \\ -0.153 & 0.108 & 0.998 & -0.744 \end{bmatrix} \quad \text{et} \quad \text{sign}(A_2) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Ces signes sont obtenus après 8 itérations de Newton. Pour la matrice A_2 on observe qu'elle est égale à l'identité. Cela indique que toutes ses valeurs propres sont à parties réelles positives, ce qui n'est pas le cas de la matrice A_1 .

2.2.2.2 Les itérations de Newton-Schultz

Cette méthode a été développée dans le but d'éviter de faire du calcul d'inverse de matrices à chaque itération. Elle se définit comme suit :

$$\begin{cases} X_{k+1} = \frac{1}{2} X_k (3I_n - X_k^2), & X_0 = A \\ \lim_{k \rightarrow \infty} (X_k) = \text{sign}(A) \end{cases} \quad (3.13)$$

Cette méthode a donc l'avantage de n'effectuer que des multiplications entre matrices, néanmoins, la convergence n'est que locale. Une condition suffisante permettant de garantir la convergence est donnée par :

$$\|I_n - A^2\| < 1 \quad (3.14)$$

Cette contrainte est décrite plus généralement dans la prochaine section sur les itérations de la famille Padé, et les preuves dans [KL91].

Exemple 2.2:

Considérons la matrice suivante :

$$A_3 = \begin{bmatrix} -3 & 2 & 4 & -2 \\ 6 & 3 & -4 & 2 \\ -2 & 1 & 8 & 3 \\ 1 & 0 & 1 & -5 \end{bmatrix}$$

La condition (3.14) n'est pas respectée, effectivement : $\|I_4 - A_3^2\| = 69.92 > 1$. Une méthode est donc de commencer par des itérations de Newton (3.11) jusqu'à ce que la condition

(3.14) soit vérifiée. Après trois itérations de Newton, c'est le cas, $\|I_4 - A_{3N}^2\| = 0.815 < 1$ avec :

$$A_{3N} = \begin{bmatrix} -0.798 & 0.445 & 0.771 & 0.079 \\ 1.455 & 0.705 & -0.675 & 0.064 \\ -0.407 & 0.125 & 1.353 & 0.631 \\ 0.036 & 0.325 & 0.239 & -0.994 \end{bmatrix}$$

On effectue ensuite les itérations de Newton-Schultz (3.13) sur la matrice A_{3N} et on obtient le signe après 6 itérations pour la même précision que l'exemple précédent:

$$\text{sign}(A_3) = \begin{bmatrix} -0.815 & 0.420 & 0.682 & 0.214 \\ 1.417 & 0.670 & -0.548 & -0.034 \\ -0.375 & 0.077 & 1.082 & 0.571 \\ -0.017 & 0.038 & 0.225 & -0.936 \end{bmatrix}$$

2.2.2.3 Les itérations de la famille Padé

Commençons par le cas scalaire. Les itérations de Padé sont dérivées de la fonction suivante, avec un scalaire $z_0 \in \mathbb{C}^*$:

$$z(t) = \tanh(t \operatorname{arctanh}(z_0)) \quad (3.15)$$

Reprenons maintenant le théorème et la preuve décrits dans [KL94].

Théorème 2.1 : *Les approximants principaux de Padé de la fonction signe sont donnés par :*

$$f_k(z) = \tanh(k \operatorname{arctanh}(z)) \quad (3.16)$$

Preuve : [KL91], le numérateur et le dénominateur polynomiaux de f_k sont, respectivement, les parties paires et impaires de $(1+z)^k$. On les notera P_k et Q_k . On note ensuite :

$$f_k(z) = \frac{P_k(z)}{Q_k(z)}, \text{ où } Q_k(z) + P_k(z) = (1+z)^k. \quad (3.17)$$

$\operatorname{arctanh}$ [GR65] s'écrit de la manière suivante :

$$\operatorname{arctanh}(z) = \frac{1}{2} \log\left(\frac{1+z}{1-z}\right) \quad (3.18)$$

Ce qui donne :

$$k \operatorname{arctanh}(z) = \log \left(\frac{1+z}{1-z} \right)^{k/2}. \quad (3.19)$$

Avec l'écriture suivante :

$$\tanh(w) = \frac{e^w - e^{-w}}{e^w + e^{-w}} \quad (3.20)$$

En combinant (3.19) et (3.20) on obtient :

$$\tanh(k \operatorname{arctanh}(z)) = \frac{(1+z)^k - (1-z)^k}{(1+z)^k + (1-z)^k}. \quad (3.21)$$

La preuve est finalement complétée en écrivant les parties paires et impaires de $(1+z)^k$:

$$P_k(z) = \frac{(1+z)^k - (1-z)^k}{2}, \quad Q_k(z) = \frac{(1+z)^k + (1-z)^k}{2} \quad (3.22)$$

■

La figure 2.1 montre une estimation de la précision de l'écriture (3.16) en fonction de k .

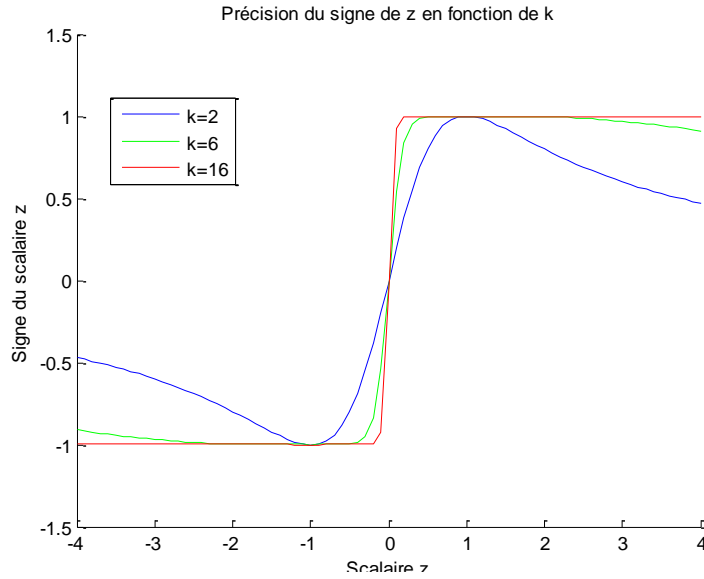


Fig. 2.1 : la fonction $y = \tanh(k \operatorname{arctanh}(x))$ pour différentes valeurs de k .

Cette figure montre clairement que plus la valeur de k est grande, plus la fonction est précise. Montrons maintenant l'évolution de ces itérations, toujours au cas scalaire. Fig. 2.2 nous montre l'évolution des itérations de la fonction (3.16) pour un scalaire donné.

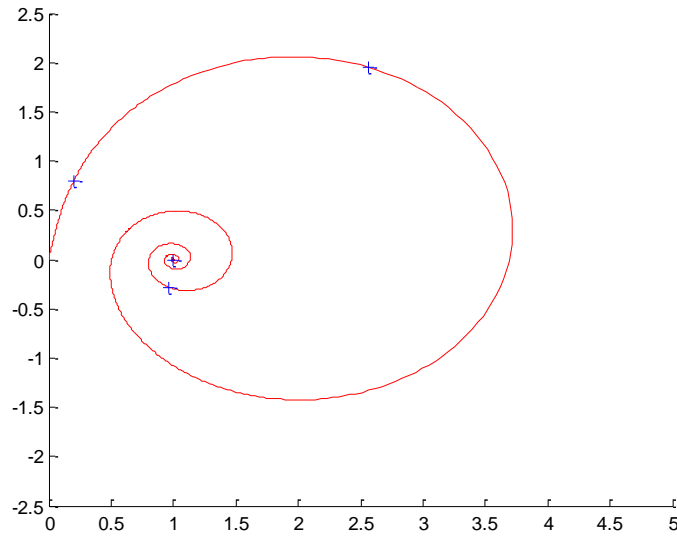


Fig. 2.2 : La spirale tanh en rouge et la séquence de Padé en bleu pour $k = 2$ et $z_0 = 0.2 + 0.8i$

Le tableau suivant, tiré de [Hig08], présente une famille de fonctions de Padé f_{lm} .

	$m = 0$	$m = 1$	$m = 2$
$l = 0$	x	$\frac{2x}{1+x^2}$	$\frac{8x}{3+6x^2-x^4}$
$l = 1$	$\frac{x}{2}(3-x^2)$	$\frac{x(3+x^2)}{1+3x^2}$	$\frac{4x(1+x^2)}{1+6x^2+x^4}$
$l = 2$	$\frac{x}{8}(15-10x^2+3x^4)$	$\frac{x}{4} \frac{(15+10x^2-x^4)}{1+5x^2}$	$\frac{x(5+10x^2+x^4)}{1+10x^2+5x^4}$

Table 2.1: Fonctions de Padé en regard de l et m

Chaque paire (l, m) peut être utilisée pour calculer le signe de x . Les itérations de Newton et de Newton-Shultz sont en fait dérivées des fonctions de Padé. On peut voir notamment pour la paire $(l=0, m=1)$ la correspondance avec l'inverse de Newton qui s'écrit $\frac{1}{2}\left(x + \frac{1}{x}\right)$. Pour la paire $(l=1, m=0)$, on a directement l'expression de Newton-Shultz.

Le cas scalaire s'étend au cas matriciel, i.e. La forme générale des approximants de Padé de la manière suivante :

$$\begin{cases} X_{k+1} = X_k \cdot p_{lm}(I_n - X_k^2) q_{lm}(I_n - X_k^2)^{-1}, & X_0 = A \\ \lim_{k \rightarrow \infty} X_k = \text{sign}(A) \end{cases} \quad (3.23)$$

$$\text{où } f_{lm}(X_k) = X_k r_{lm}(\xi) = X_k \frac{p_{lm}(\xi)}{q_{lm}(\xi)}, \quad \xi = (I_n - X_k^2)$$

Par extension de la condition de convergence pour les itérations de Newton-Shultz (3.14), on peut noter deux cas:

Pour $l \geq m-1$,

$$\text{si } \|I_n - A\| < 1, \quad \lim_{k \rightarrow \infty} X_k = \text{sign}(A) \text{ et } \|I - X_k^2\| < \|I - A^2\|^{(l+m+1)^k} \quad (3.24)$$

Pour $l = m-1$ et $l = m$,

$$(S - X_k)(S + X_k)^{-1} = \left[(S - A)(S + A)^{-1} \right]^{(l+m+1)^k}, \quad \lim_{k \rightarrow \infty} X_k = \text{sign}(A) \quad (3.25)$$

Les preuves de ces convergences sont décrites dans [KL91].

2.2.2.4 Une dérivation de l'écriture de Padé

Une manière différente de calculer la fonction signe matricielle est définie dans [Hig08], [BK95] et nommée méthode d'« expansion fractionnelle ». L'idée part des itérations de Padé au cas scalaire, données par :

$$\frac{z_{k+1} - 1}{z_{k+1} + 1} = \left(\frac{z_k - 1}{z_k + 1} \right)^\alpha \quad (3.26)$$

qui peut être réécrit de manière équivalente :

$$z_{k+1} = \frac{(z_k + 1)^\alpha + (z_k - 1)^\alpha}{(z_k + 1)^\alpha - (z_k - 1)^\alpha} \quad (3.27)$$

L'itération de Padé principale est d'ordre n_d et peut être donnée également par :

$$z_{k+1} = \tanh(\alpha \arctan z_k) \quad (3.28)$$

La fonction précédente est exprimée dans [KL95] (et dans [BK95]) comme une somme de fonctions partielles pour le cas pair $\alpha = 2d$:

$$\tanh(2d \operatorname{arctanh}(z)) = \frac{z}{\alpha} \sum_{m=0}^{\alpha-1} \frac{1}{\sin^2\left(\frac{(2m+1)\pi}{4d}\right) + \cos^2\left(\frac{(2m+1)\pi}{4d}\right) z^2} \quad (3.29)$$

Et pour le cas impair $\alpha = 2d + 1$:

$$\tanh\left((2d+1) \operatorname{arctanh}(z)\right) = \frac{z}{2d+1} + \frac{2z}{2d+1} \sum_{m=0}^{\alpha-1} \frac{1}{\sin^2\left(\frac{(2m+1)\pi}{2(2d+1)}\right) + \cos^2\left(\frac{(2m+1)\pi}{2(2d+1)}\right) z^2} \quad (3.30)$$

En utilisant l'expansion fractionnelle partielle de (3.29) nous obtenons une itération parallèle pour le cas pair [PKL90] étendue au cas matriciel :

$$\begin{cases} Z_{k+1} = \frac{1}{\alpha} Z_k \sum_{m=1}^{\alpha} \left(\sin^2\left(\frac{(2m+1)\pi}{4\alpha}\right) I_n + \cos^2\left(\frac{(2m+1)\pi}{4\alpha}\right) Z_k^2 \right)^{-1} \\ Z_0 = Z \end{cases} \quad (3.31)$$

où $\lim_{k \rightarrow \infty} Z_k = \operatorname{sign}(Z)$. Il existe une autre manière d'atteindre $\operatorname{sign}(Z)$ sans faire varier k . Il s'agit en effet d'une suite de sommes dépendantes de α . La convergence est donc également assurée pour $\lim_{\alpha \rightarrow \infty} Z_1 = \operatorname{sign}(Z)$.

2.2.2.5 *Scaling* sur les itérations de Newton

Le *scaling* est une méthode d'échelonnage qui permet d'obtenir une convergence plus rapide des itérations. Pour Newton, X_k est remplacé par $\mu_k X_k$. Les itérations sont notées comme suit [Hig08]:

$$\begin{cases} X_{k+1} = \frac{1}{2} (\mu_k X_k + \mu_k^{-1} X_k^{-1}), X_0 = A \\ \lim_{k \rightarrow \infty} X_k = \operatorname{sign}(A) \end{cases} \quad (3.32)$$

Le critère de *scaling* μ_k est réel et positif pour ne pas changer le signe de A . De nombreux types de *scaling* existent, dont les suivants :

$$\text{Basé sur le déterminant : } \mu_k = |\det(X_k)|^{-1/n} \quad (3.33)$$

$$\text{Basé sur le spectre : } \mu_k = \sqrt{\rho(X_k^{-1}) \rho(X_k)^{-1}} \quad (3.34)$$

$$\text{Basé sur la norme : } \mu_k = \sqrt{\|X_k^{-1}\| \cdot \|X_k\|^{-1}} \quad (3.35)$$

Exemple 2.3:

Reprenons la matrice :

$$A_1 = \begin{bmatrix} 1 & 2 & 4 & -2 \\ 6 & 3 & -4 & 2 \\ -2 & 1 & 8 & 3 \\ 1 & 0 & 7 & -5 \end{bmatrix}$$

Appliquons les itérations (3.32) pour chaque *scaling* proposé. On aura pour le cas du déterminant : $\mu_0 = 0.203$, le cas du spectre : $\mu_0 = 0.213$ et le cas de la norme : $\mu_0 = 0.265$. Le résultat obtenu est identique à celui des itérations de Newton, et pour la même précision, les trois méthodes de *scaling* ont nécessité 5 itérations.

2.2.2.6 Les itérations dérivées des polynômes de Laurent

2.2.2.6.1 Définition

Il s'agit là d'une méthode se basant sur les travaux de [IM11]. Justifions dans un premier temps l'utilisation de la formule du signe suivante :

$$\text{sign}(A) = A(A^2)^{-\frac{1}{2}}, \quad A \in \mathbb{C}_{n \times n}^* \quad (3.36)$$

On présente ici deux méthodes existantes pour la preuve de cette formule.

Première méthode : D'après [Hig08], la décomposition matricielle du signe s'écrit :

$$A = SN \quad (3.37)$$

Et cela implique :

$$N = (A^2)^{\frac{1}{2}} \quad (3.38)$$

Preuve :

$$N = S^{-1}A = SA \quad \text{car } S^2 = I, \text{ propriété du signe.}$$

On sait également que A commute avec son signe S , c.à.d. : $SA = AS$. Il en découle ce qui suit:

$$N^2 = NN = SAS^{-1}A = ASS^{-1}A = A^2$$

On a finalement la propriété suivante :

$$\lambda(SA) \subset \mathbb{C}^{*+}$$

$$\text{Donc : } N = (A^2)^{\frac{1}{2}} \text{ existe, et } A = S(A^2)^{\frac{1}{2}} \Rightarrow S = A(A^2)^{-\frac{1}{2}}. \quad \blacksquare$$

Deuxième méthode : On cherche la solution anti-stabilisante de l'équation de Riccati (non symétrique) :

$$A - SAS = 0 \quad (3.39)$$

Le signe de A est solution de cette équation et $\lambda(AS) \in \mathbb{C}^{*+}$.

Preuve :

On sait que :

$$\begin{pmatrix} 0 & A \\ A & 0 \end{pmatrix} \begin{pmatrix} I & 0 \\ S & -I \end{pmatrix} = \begin{pmatrix} I & 0 \\ S & -I \end{pmatrix} \begin{pmatrix} AS & -A \\ 0 & -SA \end{pmatrix}$$

Donc :

$$\text{sign} \begin{pmatrix} 0 & A \\ A & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ S & -I \end{pmatrix} \text{sign} \begin{pmatrix} AS & -A \\ 0 & -SA \end{pmatrix} \begin{pmatrix} I & 0 \\ S & -I \end{pmatrix}$$

On introduit \bar{X} telle que : $\text{sign} \begin{pmatrix} AS & -A \\ 0 & -SA \end{pmatrix} = \begin{pmatrix} I & \bar{X} \\ 0 & -I \end{pmatrix}$.

Et on obtient :

$$\begin{aligned} \text{sign} \begin{pmatrix} 0 & A \\ A & 0 \end{pmatrix} &= \begin{pmatrix} I & 0 \\ S & -I \end{pmatrix} \begin{pmatrix} I & \bar{X} \\ 0 & -I \end{pmatrix} \begin{pmatrix} I & 0 \\ S & -I \end{pmatrix} \\ &= \begin{pmatrix} I + \bar{X}S & -\bar{X} \\ 2S + S\bar{X}S & -S\bar{X} - I \end{pmatrix} \end{aligned}$$

Or, il est trivial de constater que :

$$\text{sign} \begin{pmatrix} 0 & A \\ A & 0 \end{pmatrix} = \begin{pmatrix} 0 & * \\ * & 0 \end{pmatrix}$$

Ce qui implique : $\begin{cases} I + \bar{X}S = 0 \\ \bar{X} = -S \end{cases}$

D'où finalement : $\text{sign}(A) = S$. ■

2.2.2.6.2 Les polynômes de Laurent pour le calcul de la fonction signe

L'expression de la matrice polynomiale de Laurent initiale est donnée par [IM11] pour $z \in \mathbb{C}^*$:

$$L(z) = Pz^{-1} + Q + Pz \quad (3.40)$$

Afin de faire le lien avec la fonction signe, voire même avec d'autres fonctions, posons $P = \frac{1}{4}(S - T)$ et $Q = \frac{1}{2}(S + T)$. En fonction des valeurs que prendront S et T , le coefficient constant \bar{F}_0 de l'inverse du polynôme noté $L(z)^{-1}$ donnera directement la fonction voulue (matrice signe, racine matricielle...). \bar{F}_0 est donné par :

$$\bar{F}_0 = (I - 4M^2)^{-\frac{1}{2}} Q^{-1} \quad (3.41)$$

avec $M = Q^{-1}P$. Il est nécessaire de noter que le domaine d'inversibilité de $L(z)$ est donné par l'anneau :

$$A_R = \{R < |z| < 1/R\} \quad (3.42)$$

et R est défini par :

$$R = \rho \left(-2M \left(I + (I - 4M^2)^{1/2} \right)^{-1} \right) \quad (3.43)$$

Ou encore, en remplaçant Q et M définis précédemment on obtient :

$$\bar{F}_0 = 2 \left(I - \left((S + T)^{-1} (S - T) \right)^2 \right)^{-\frac{1}{2}} (S + T)^{-1} \quad (3.44)$$

Nous introduisons maintenant la notation suivante en reprenant (3.40) :

$$L(z; S, T) = \frac{1}{4}(S - T)z^{-1} + \frac{1}{2}(S + T) + \frac{1}{4}(S - T)z \quad (3.45)$$

et

$$\xi(V) = \rho \left((V - I)(V + I)^{-1} \right), \quad (3.46)$$

où V est une matrice telle que $V + I$ est non singulière, et $\rho(\cdot)$ étant le rayon spectral. Le prochain théorème, Théorème 8 de [IM11], concerne la représentation de la racine matricielle $A^{\frac{1}{2}}$ et de la fonction signe $\text{sign}(A)$ en fonction du coefficient constant \bar{F}_0 de la matrice polynomiale de Laurent $L(z; S, T)^{-1}$.

Théorème 2.2: *Les deux propriétés suivantes sont vraies :*

1. Si A est une matrice n'ayant pas de valeurs propres non-positives, $S = I$ et $T = A^{-1}$, alors $L(z; S, T)$ est inversible dans A_R , où $R = \xi \left(A^{-\frac{1}{2}} \right)$, et $\bar{F}_0 = A^{-\frac{1}{2}}$.
2. Si A est une matrice n'ayant pas de valeurs propres imaginaires, $S = A^{-1}$ et $T = A$, alors $L(z; S, T)$ est inversible dans A_R , où $R = \xi \left((A^2)^{\frac{1}{2}} \right)$, et $\bar{F}_0 = \text{sign}(A)$.

Preuve : La première propriété a été prouvée dans [Mei04]. La seconde propriété se déduit de la première étant donné que $A^{-1}L(z; I, A^2) = L(z; A^{-1}, A)$; le coefficient constant de $L(z; I, A^2)^{-1}$ est $(A^{-2})^{\frac{1}{2}}$, donc, le coefficient constant de $L(z; A^{-1}, A)^{-1}$ est $A(A^{-2})^{\frac{1}{2}} = A(A^2)^{-\frac{1}{2}} = \text{sign}(A)$. ■

D'autres résultats existent et sont résumés dans le tableau suivant :

\bar{F}_0	S	T
$A^{\frac{1}{2}}$	I	A^{-1}
$\text{sign}(A)$	A^{-1}	A
$\text{polar}(A)$	A^{-1}	A^*
$A \# B$	A^{-1}	B^{-1}

Table 2.2: Le coefficient \bar{F}_0 de $L(z; S, T)^{-1}$

avec $\text{polar}(A)$ étant le facteur polaire d'une matrice non singulière et $A \# B$ le sens géométrique de deux matrices positives.

2.3 Comparaison des méthodes itératives au cas constant

Afin d'avoir un aperçu des différentes méthodes présentées dans ce chapitre, nous proposons dans ce qui suit une étude comparative. Dans cette partie nous considérerons les méthodes suivantes :

- Les itérations de Newton
- Les itérations de Newton-Shultz
- Les itérations de Padé pour les paires $(l=1, m=1)$ et $(l=2, m=2)$.
- Les itérations de Newton via *scaling* :
 - Cas du déterminant
 - Cas du spectre

○ Cas de la norme

Les itérations de Newton-Shultz nécessitant également une condition, nous effectuons donc premièrement les itérations de Newton classiques jusqu'à ce que la condition pour les itérations de Newton-Shultz soit vérifiée.

Pour une grande série de tests sur des matrices aléatoires, nous présentons les résultats obtenus pour chaque méthode itérative selon le nombre d'itérations nécessaires pour une précision ε donnée et le temps de calcul de chaque méthode.

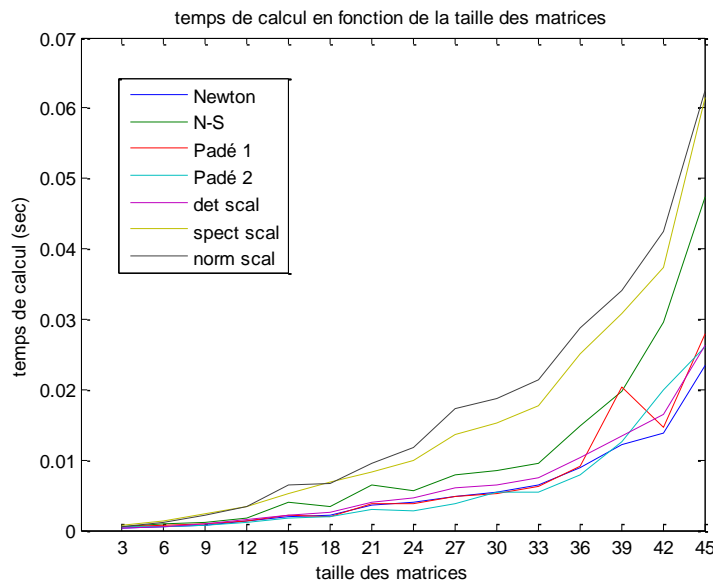


Fig. 2.3: Temps de calcul en fonction de la taille des matrices

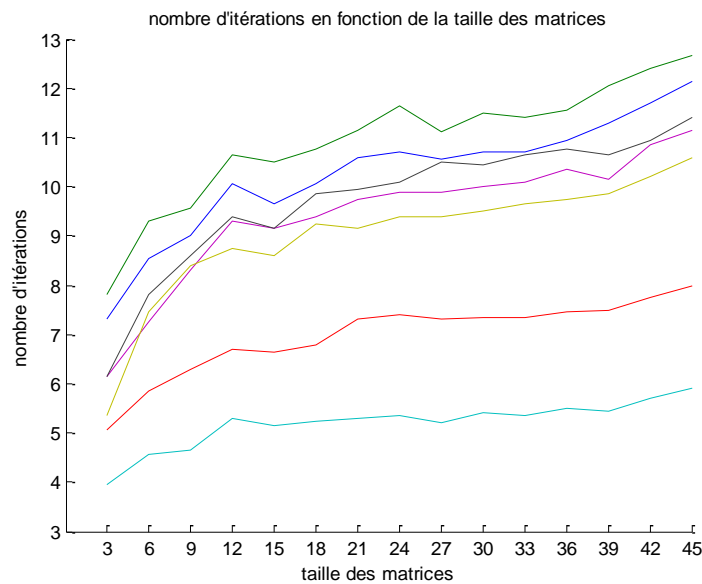


Fig. 2.4: Nombre d'itérations en fonction de la taille des matrices

La figure 2.3 montre l'évolution du temps de calcul de la fonction signe matricielle en fonction de la taille des matrices. La figure 2.4 montre l'évolution du nombre d'itérations nécessaires pour atteindre la précision $\varepsilon = 10^{-8}$ en fonction également de la taille des matrices. Pour chaque taille de matrice, on effectue 10 tests de matrices aléatoires, afin de « lisser » les courbes. On remarquera que pour les deux cas, les différences sont quasi-négligeables entre les méthodes au cas constant. Néanmoins, plus les matrices sont grandes, plus la possibilité de se retrouver avec des valeurs propres proche de l'axe imaginaire sont grandes. Dans ce cas, les convergences peuvent être beaucoup plus longue, voir mener à un résultat erroné. Il est préférable d'utiliser par exemple le signe étendu, développé dans la section 4.3. En terme de complexité, les méthodes sont toutes approximées en $O(k.n^3)$, où k est le nombre d'itérations. Les différences se font dans le nombre de multiplications et d'inversions de chaque méthode, les itérations de Padé montrées précédemment sont par exemple plus « gourmandes » en nombre d'opérations, mais convergent plus rapidement. Ici un tableau de comparaison entre ces différentes méthodes est également présenté.

Précision $\varepsilon = 10^{-8}$	Itérations Newton	Itérations Newton- Shultz	Itérations Padé	Scaling Déterminant	Scaling Spectre	Scaling Norme
Matrices aléatoires $n = 50$	13.85	14.37	9.4	10.7	10.2	11.1
	0.032	0.059	0.036	0.031	0.074	0.085
Matrice 'Lotkin' $n = 10$	48	49	31	11	8	10
	0.0039	0.0067	0.0054	0.0013	0.0041	0.0031
Matrice 'Grcar' $n = 50$	12	12	8	10	10	34
	0.014	0.021	0.015	0.015	0.045	0.152
Matrices à v.p. : $\lambda(Z) = \{1, 1, 1, 1, 1000\}$	18	18.6	12	9	3	3
	0.001	0.0014	0.0012	0.0006	0.0007	0.0004
Matrices à v.p. : $\lambda(Z) = \{1, 1, 1, 1, 1 + 1000i\}$	25	25	16	17	8	8
	0.0015	0.002	0.0019	0.0011	0.0012	0.001

Table 2.3: Comparaison des différentes méthodes de calcul de la fonction signe matricielle

Ce tableau a pour objectif de faire une comparaison entre les différentes méthodes de calcul de la fonction signe matricielle sur différents types de matrices, en vue de montrer les avantages et inconvénients des différentes méthodes présentées. Pour chaque matrice étudiée, la première ligne correspond au nombre d'itérations nécessaire pour la convergence, pour une précision de $\varepsilon = 10^{-8}$. La seconde ligne montre le temps moyen de

calcul. Chaque type de matrices étudié, en dehors des matrices ‘Lotkin’ et ‘Grcar’, montre une moyenne sur 100 matrices.

La matrice dite ‘Lotkin’ de taille $n=10$ provient de la galerie de matrices dans Matlab, `gallery('lotkin', 10)`. Il s’agit d’une matrice mal conditionnée contenant beaucoup de valeurs propres négatives, de faible magnitude. Plus la taille de la matrice est grande, plus on aura des valeurs propres proches de 0. Cette matrice est présentée afin de montrer l’efficacité des méthodes de *scaling*, elles ont en effet besoin de beaucoup moins d’itérations pour converger.

La matrice dite ‘Grcar’ de taille $n=50$ provient également de la galerie de matrices dans Matlab, `gallery('grcar', 50)`. Il s’agit d’une matrice de Toeplitz avec des valeurs propres sensibles, toutes les valeurs propres sont complexes conjuguées. Cette matrice est présentée afin de montrer qu’elle pose problème dans le cas du *scaling* de norme. Elle est dans ce cas pire en termes d’itérations que la méthode de Newton classique.

Les deux derniers types de matrices ont des valeurs propres égales à 1 sauf pour une valeur propre grande, réelle dans l’avant dernier cas et imaginaire dans le dernier. Ces deux derniers types de matrices sont présentés afin de montrer la difficulté pour le *scaling* de déterminant. En effet, même si elle nécessite moins d’itérations que la méthode de Newton classique, elle a besoin de beaucoup plus d’itérations que les autres méthodes de *scaling*.

On remarque que la méthode de Newton-Shultz est la plus gourmande généralement en terme d’itérations. Cela s’explique par le fait qu’elle n’utilise aucune inversion de matrices. Néanmoins, on verra par la suite que cette méthode est préférable dans certains cas lorsque nous aborderons le cas de matrices paramétrées. Les itérations de Padé sont en quelque sorte des itérations de Newton d’ordre supérieur, c’est pourquoi elles convergent plus vite en terme d’itérations mais nécessitent globalement plus de temps de calcul que les itérations de Newton classiques. Finalement, on a montré que les méthodes de *scaling* nécessitent moins d’itérations que les autres méthodes, sauf dans certains cas précis, comme montrés dans les paragraphes précédents. Il est donc nécessaire de bien connaître la matrice de départ afin de savoir si ces méthodes sont préférables aux itérations de Newton classiques, ce qui n’est pas évident au cas paramétré.

Pour conclure sur ce tableau, on peut observer que pour chaque type de matrices la convergence est assurée pour les différentes méthodes proposées. Tous les temps de calcul de chaque méthode sont proches et ne dépassent généralement pas le dixième de seconde.

2.4 Fonction signe matricielle pour l’automatique

Cette partie a pour but d’éveiller les sens de l’automaticien. En introduisant la fonction signe matricielle pour des problèmes d’automatique. On y montrera la relation directe

existante avec la stabilité robuste, puis ses liens avec les équations de Riccati, de Lyapunov et de Sylvester.

2.4.1 Liens entre la fonction signe matricielle et la stabilité robuste

Une manière d'étudier la stabilité d'une matrice repose sur le calcul de sa matrice signe correspondante. En effet, nous avons l'équivalence suivante :

$$\text{sign}(Z) = -I_n \Leftrightarrow \lambda(Z) \in \mathbb{C}_- \quad (3.47)$$

Notons que cette méthode n'est pas seulement utile pour évaluer la stabilité (robuste) d'un système LTI (LTI incertain) mais peut également servir à caractériser l'espace paramétrique exact de stabilité pour une matrice $Z(\theta)$ (qui peut être convexe ou même contenir des ensembles disjoints).

$$\text{sign}(Z(\theta)) = -I_n \Leftrightarrow \lambda(Z(\theta)) \in \mathbb{C}_- \quad (3.48)$$

En fait, $\text{sign}(Z(\theta)) = -I_n$ mène à un ensemble d'équations non linéaires qui caractérisent le domaine de la stabilité robuste.

Exemple 2.4:

Reprenons les deux matrices A_1 et $\tilde{A}_2 = -A_2$ constantes suivantes :

$$A_1 = \begin{bmatrix} 1 & 2 & 4 & -2 \\ 6 & 3 & -4 & 2 \\ -2 & 1 & 8 & 3 \\ 1 & 0 & 7 & -5 \end{bmatrix}, \tilde{A}_2 = -\begin{bmatrix} 1 & 2 & 3 & -2 \\ 0 & 3 & -4 & 2 \\ 1 & 1 & 8 & 3 \\ 2 & 0 & 7 & 5 \end{bmatrix}$$

Nous avons trouvé leurs signes correspondant :

$$\text{sign}(A_1) = \begin{bmatrix} -0.398 & 0.570 & 0.513 & 0.091 \\ 1.837 & 0.255 & -0.599 & -0.258 \\ -0.374 & 0.140 & 0.887 & 0.489 \\ -0.153 & 0.108 & 0.998 & -0.744 \end{bmatrix}, \text{sign}(\tilde{A}_2) = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}.$$

Cela montre que la matrice \tilde{A}_2 est stable (toutes ses valeurs propres sont à partie réelles négatives) car $\text{sign}(\tilde{A}_2)$ est égal à $-I_4$, ce qui n'est pas le cas de la matrice A_1 .

En effet, leurs valeurs propres respectives sont : $\lambda(A_1) = [-4.8005, -2.5141, 5.548, 8.7666]$ et $\lambda(\tilde{A}_2) = [-11.223, -3.0721, -1.3522 \pm 2.0075i]$.

Pour aller plus loin, nous pouvons également montrer le lien direct avec les systèmes bidimensionnels (2D). On rappellera que ces systèmes ont été introduits dans les années 70 par [FM76] et [GR72]. On rappelle qu'un système bidimensionnel continu-continu, à titre d'exemple, peut s'écrire sous la forme suivante :

$$\begin{bmatrix} \frac{\partial x^h(t_1, t_2)}{\partial t_1} \\ \frac{\partial x^v(t_1, t_2)}{\partial t_2} \end{bmatrix} = Ax(t_1, t_2) + Bu(t_1, t_2) \quad (3.49)$$

avec: $\begin{cases} x^h(0, t_2) = f(t_2) \\ x^v(t_1, 0) = g(t_1) \end{cases}$ et $x(t_1, t_2) = \begin{bmatrix} x^h(t_1, t_2) \\ x^v(t_1, t_2) \end{bmatrix}$.

Ici, $x^h(t_1, t_2) \in \mathbb{R}^{n_1}$ correspond à l'état horizontal, $x^v(t_1, t_2) \in \mathbb{R}^{n_2}$ correspond à l'état vertical. Enfin, $f(t_2)$ et $g(t_1)$ correspondent aux conditions limites. On suppose que la matrice d'état est constante et se décompose de la manière suivante :

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad (3.50)$$

Pour analyser la stabilité de ces systèmes, on peut utiliser la condition de stabilité basée sur le calcul du polynôme caractéristique :

$$C(s_1, s_2) \neq 0, \forall (s_1, s_2) : \operatorname{Re}(s_1) \geq 0, \operatorname{Re}(s_2) \succ 0,$$

$$\text{Où : } C(s_1, s_2) = \det \begin{pmatrix} s_1 I_{n_1} - A_{11} & -A_{12} \\ -A_{21} & s_2 I_{n_2} - A_{22} \end{pmatrix}$$

D'où, la condition nécessaire et suffisante de stabilité suivante :

Le système est stable, si et seulement si le polynôme caractéristique est un polynôme de Hurwitz. Afin d'étudier la stabilité de ces systèmes en utilisant la fonction signe matricielle on notera que l'on peut se ramener à l'écriture suivante :

$$x_h(t_2) = G(s_1) x_h(t_2)$$

avec une matrice d'état : $G(s) = A_{22} + A_{21}(sI - A_{11})^{-1} A_{12}$ dépendant du paramètre scalaire complexe s_1 [Aga90]. A partir de ce point l'utilisation de la fonction signe d'une matrice paramétrée devient évidente.

2.4.2 Liens entre la fonction signe matricielle et l'équation de Riccati

Considérons l'équation algébrique de Riccati :

$$A^T X + XA - XGX + Q = 0 \quad (3.51)$$

pour laquelle la paire (A, G) est supposée stabilisable et la paire (A, Q) détectable. Cette équation peut être résolue en utilisant la matrice Hamiltonienne :

$$H = \begin{bmatrix} A & -G \\ -Q & -A^T \end{bmatrix} \in \mathbb{R}^{2n \times 2n} \quad (3.52)$$

Hypothèse : La matrice Hamiltonienne H n'a pas de valeurs propres sur l'axe imaginaire.

Théorème 2.3 :

Notons :

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} = \text{sign}(H) \quad (3.53)$$

Il existe une unique matrice symétrique semi-définie positive X_p satisfaisant :

$$\begin{bmatrix} S_{12} \\ S_{22} + I \end{bmatrix} X_p = - \begin{bmatrix} S_{11} + I \\ S_{21} \end{bmatrix} \quad (3.54)$$

De plus, X_p est l'unique solution positive de l'équation de Riccati (3.51).

Lemme 2.1:

Soit T une matrice définie par :

$$T = \begin{bmatrix} I & Y \\ 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ -X & I \end{bmatrix} = \begin{bmatrix} I - YX & Y \\ -X & I \end{bmatrix} \quad (3.55)$$

L'égalité suivante est vérifiée :

$$THT^{-1} = \begin{bmatrix} (A - GX) & 0 \\ 0 & -(A - GX)^T \end{bmatrix} \quad (3.56)$$

Preuve : Notons les deux équations suivantes :

$$\begin{aligned} A^T X + XA - XGX + Q &= 0 \\ (A - GX)Y + Y(A - GX)^T &= -G \end{aligned}$$

L'inverse de T s'écrit :

$$T^{-1} = \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \begin{bmatrix} I & -Y \\ 0 & I \end{bmatrix} = \begin{bmatrix} I & -Y \\ X & I - XY \end{bmatrix}$$

d'où :

$$THT^{-1} = \begin{bmatrix} I - YX & Y \\ -X & I \end{bmatrix} \begin{bmatrix} A & -G \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} I & -Y \\ X & I - XY \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{bmatrix}$$

avec :

$$\begin{aligned} \alpha_{11} &= A - YXA - YQ - GX + YXGX - YA^T X \\ &= A - GX - Y(A^T X + XA - XGX + Q) = A - GX \\ \alpha_{12} &= -AY + YXAY + YQY - G + YXG - YA^T + GXY - YXGXY + YA^T XY \\ &= -\left((A - GX)Y + Y(A - GX)^T + G\right) + Y(A^T X + XA - XGX + Q)Y = 0 \\ \alpha_{21} &= -(A^T X + XA - XGX + Q) = 0 \\ \alpha_{22} &= XAY + QY + XG - A^T + XGXY + A^T XY \\ &= -(A - GX)^T + (A^T X + XA - XGX + Q)Y = (A - GX)^T \end{aligned}$$

■

Cela mène donc finalement à (3.56).

Éléments de preuve du Théorème 2.3 :

Reprenons la définition de la fonction signe donnée par (3.2) et utilisant la matrice $Z = T^{-1}$ introduite dans le lemme précédent:

$$\begin{aligned} \text{sign}(H) &= T^{-1} \begin{bmatrix} -I & 0 \\ 0 & I \end{bmatrix} T \\ &= \begin{bmatrix} I & -Y \\ X & I - XY \end{bmatrix} \begin{bmatrix} -I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I - YX & Y \\ -X & I \end{bmatrix} = \begin{bmatrix} -I + 2YX & -2Y \\ -2X + 2XYX & I - 2XY \end{bmatrix} \end{aligned}$$

Nous pouvons adopter la partition suivante et procéder à l'identification:

$$\text{sign}(H) + I = \begin{bmatrix} S_{11} + I & S_{12} \\ S_{21} & S_{22} + I \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 2Y \\ 2(XY - I) \end{bmatrix} X & -\begin{bmatrix} 2Y \\ 2(XY - I) \end{bmatrix} \end{bmatrix}$$

Ceci conduit à :

$$\begin{cases} \begin{bmatrix} S_{11} + I \\ S_{21} \end{bmatrix} = \begin{bmatrix} 2Y \\ 2(XY - I) \end{bmatrix} X \\ \begin{bmatrix} S_{12} \\ S_{22} + I \end{bmatrix} = - \begin{bmatrix} 2Y \\ 2(XY - I) \end{bmatrix} \end{cases}$$

Finalement, en remplaçant la seconde égalité dans la première, (3.54) est retrouvée. ■

Exemple 2.5:

Considérons le système suivant :

$$\Sigma_1 : \begin{cases} A_1 = \begin{bmatrix} 1 & 2 & 4 & -2 \\ 6 & 3 & -4 & 2 \\ -2 & 1 & 8 & 3 \\ 1 & 0 & 7 & -5 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, C = [1 \ 0 \ -1 \ 0] \\ Q = C^T C, R = 1, G = BR^{-1}B^T. \end{cases}$$

Pour résoudre l'équation de Riccati de la forme (3.51), la forme Hamiltonienne correspondante s'écrit :

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} = \begin{bmatrix} 1 & 2 & 4 & -2 & 0 & 0 & 0 & 0 \\ 6 & 3 & -4 & 2 & 0 & 0 & 0 & 0 \\ -2 & 1 & 8 & 3 & 0 & 0 & 0 & 0 \\ 1 & 0 & 7 & -5 & 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 0 & -1 & -6 & 2 & -1 \\ 0 & 0 & 0 & 0 & -2 & -3 & -1 & 0 \\ 1 & 0 & -1 & 0 & -4 & 4 & -8 & -7 \\ 0 & 0 & 0 & 0 & 2 & -2 & -3 & 5 \end{bmatrix}$$

Calculons le signe de H en utilisant les itérations de Newton classique, et pour une précision de l'ordre de 10^{-8} . Après 8 itérations nous obtenons :

$$\text{sign}(H) = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} =$$

$$= \begin{bmatrix} -0.388 & 0.5628 & 0.5139 & 0.087 & -0.0907 & 0.1214 & -0.0314 & 0.0131 \\ 1.8216 & 0.2647 & -0.6005 & -0.2516 & 0.1214 & -0.1643 & 0.0468 & -0.0241 \\ -0.3672 & 0.1377 & 0.8868 & 0.4859 & -0.0314 & 0.0468 & -0.0242 & 0.0280 \\ -0.1605 & 0.1083 & 1.001 & -0.7404 & 0.0131 & -0.0241 & 0.0280 & -0.1062 \\ -0.2371 & 0.0591 & 0.0089 & 0.0821 & 0.388 & -1.8216 & 0.3672 & 0.1605 \\ 0.0591 & -0.0411 & 0.0051 & -0.0199 & -0.5628 & -0.2647 & -0.1377 & -0.1083 \\ 0.0895 & 0.0051 & -0.0723 & -0.0188 & -0.5139 & 0.6005 & -0.8868 & -1.001 \\ 0.0821 & -0.0199 & -0.0188 & -0.0521 & -0.087 & 0.2516 & -0.4859 & 0.7404 \end{bmatrix}$$

Maintenant il faut résoudre l'équation linéaire $\begin{bmatrix} S_{12} \\ S_{22} + I \end{bmatrix} X = -\begin{bmatrix} S_{11} + I \\ S_{21} \end{bmatrix}$.

On obtient alors la solution de l'équation de Riccati avec une précision de l'ordre de 10^{-7} :

$$X = \begin{bmatrix} 18921 & 15742 & 6372.9 & 449.61 \\ 15742 & 13111 & 5367.9 & 390.35 \\ 6342.9 & 5367.9 & 2440.8 & 224.13 \\ 449.61 & 390.35 & 224.13 & 28.655 \end{bmatrix}$$

2.4.3 Liens entre la fonction signe matricielle et l'équation de Lyapunov

Considérons maintenant l'équation algébrique de Lyapunov donnée par :

$$A^T X + XA + Q = 0 \quad (3.57)$$

Théorème 2.4 : [KL95]

La solution de l'équation de Lyapunov (3.57) peut être déduite de la matrice Hamiltonienne :

$$H = \begin{bmatrix} A & 0 \\ -Q & -A^T \end{bmatrix} = \begin{bmatrix} S_{11} & 0 \\ S_{21} & S_{22} \end{bmatrix} \quad (3.58)$$

grâce à l'égalité suivante :

$$\text{sign}(H) = \begin{bmatrix} -I & 0 \\ 2X & I \end{bmatrix} \quad (3.59)$$

La solution finale est donc :

$$X = \frac{1}{2} S_{21} \quad (3.60)$$

Preuve : La preuve découle directement de celle du Théorème 2.3 en prenant $G = 0$. ■

Exemple 2.6:

Considérons le système suivant :

$$\Sigma_2 : \begin{cases} A = \begin{bmatrix} 1 & 2 \\ 1 & 5 \end{bmatrix}, C = [1 \quad 0] \\ Q = C' C \end{cases}$$

Nous voulons résoudre l'équation de Lyapunov de la forme (3.57). Pour cela, nous commençons par écrire la forme Hamiltonienne correspondante au problème :

$$H = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 1 & 5 & 0 & 0 \\ -1 & 0 & -1 & -1 \\ 0 & 0 & -2 & -5 \end{bmatrix}$$

Calculons le signe de H . Nous utilisons les itérations de Newton classique, et pour une précision de l'ordre de 10^{-8} . Après 7 itérations nous obtenons :

$$\text{sign}(H) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1.556 & 0.556 & -1 & 0 \\ 0.556 & -0.222 & 0 & -1 \end{bmatrix}$$

Nous obtenons maintenant la solution de l'équation de Lyapunov (3.57) en utilisant (3.60) :

$$X = \begin{bmatrix} -0.778 & 0.278 \\ 0.278 & -0.111 \end{bmatrix}$$

Considérons également l'équation de Lyapunov généralisée :

$$A^T X E + E^T X A + Q = 0 \quad (3.61)$$

Théorème 2.5 : [KL95]

Notons :

$$E_1 = \begin{bmatrix} I & 0 \\ 0 & E^T \end{bmatrix}, E_2 = \begin{bmatrix} E & 0 \\ 0 & I \end{bmatrix} \quad (3.62)$$

La fonction signe généralisée est définie par :

$$gsign(H) = E_1 sign(E_1^{-1} H E_2^{-1}) E_2 \quad (3.63)$$

L'équation qui en découle est la suivante :

$$gsign(H) = \begin{bmatrix} S_{11} & 0 \\ S_{21} & S_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & E^T \end{bmatrix} \begin{bmatrix} -I & 0 \\ -2X & I \end{bmatrix} \begin{bmatrix} E & 0 \\ 0 & I \end{bmatrix} = \begin{bmatrix} -E & 0 \\ -2E^T X E & E^T \end{bmatrix} \quad (3.64)$$

Ce résultat nous permet de résoudre l'équation généralisée de Lyapunov (3.61), grâce à :

$$\begin{aligned} S_{11} &= -E, \quad S_{22} = E^T \\ X &= \frac{1}{2} S_{22}^{-1} S_{21} S_{11}^{-1} \end{aligned} \quad (3.65)$$

Exemple 2.7:

Considérons le système suivant :

$$\Sigma_3 : \begin{cases} A = \begin{bmatrix} 1 & 2 \\ 1 & 5 \end{bmatrix}, C = [1 \quad 0], E = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} \\ Q = C^T C \end{cases}$$

Nous voulons résoudre l'équation de Lyapunov de la forme (3.61). Pour cela, nous commençons par écrire la forme Hamiltonienne correspondante au problème :

$$H = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 1 & 5 & 0 & 0 \\ -1 & 0 & -1 & -1 \\ 0 & 0 & -2 & -5 \end{bmatrix}$$

Il faut maintenant calculer la forme généralisée du signe de H :

$$gsign(H) = \begin{bmatrix} I_2 & 0 \\ 0 & E^T \end{bmatrix} sign \left(\begin{bmatrix} -3 & 2 & 0 & 0 \\ -9 & 5 & 0 & 0 \\ -1 & 0 & 3 & 9 \\ 0 & 0 & -2 & -5 \end{bmatrix} \right) \begin{bmatrix} E & 0 \\ 0 & I_2 \end{bmatrix}$$

Nous utilisons les itérations de Newton classique, et pour une précision de l'ordre de 10^{-8} . Après 7 itérations nous obtenons :

$$g\text{sign}(H) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ -0.667 & 0.333 & -1 & -2 \\ 0.333 & -0.667 & 0 & -1 \end{bmatrix}$$

Nous obtenons maintenant la solution de l'équation de Lyapunov généralisée (3.61) en utilisant (3.65) :

$$X = \begin{bmatrix} -2.333 & 0.833 \\ 0.833 & -0.333 \end{bmatrix}$$

2.4.4 Liens entre la fonction signe matricielle et l'équation de Sylvester

Dans cette partie, nous montrerons que la relation entre l'équation de Sylvester et la fonction signe matricielle peut être établie de la même façon que pour les équations de Riccati ou de Lyapunov. L'équation de Sylvester s'écrit :

$$BX - XA = C \quad (3.66)$$

avec $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times m}$ et $X \in \mathbb{R}^{m \times n}$,

Considérons l'équation de Sylvester de la forme (3.66) et la matrice Hamiltonienne associée donnée par :

$$H = \begin{bmatrix} -A & 0 \\ C & -B \end{bmatrix} \quad (3.67)$$

Posons pour la matrice signe H :

$$\text{sign}(H) = \begin{bmatrix} S_{11} & 0 \\ S_{21} & S_{22} \end{bmatrix} \quad (3.68)$$

Selon [Rob80], la matrice H peut être bloc-diagonalisée en utilisant la transformation $\begin{bmatrix} I & 0 \\ X & I \end{bmatrix}$. Le calcul de la fonction signe matricielle de H donne :

$$\text{sign}(H) = \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \text{sign} \left(\begin{bmatrix} -A & 0 \\ 0 & -B \end{bmatrix} \right) \begin{bmatrix} I & 0 \\ X & I \end{bmatrix}^{-1} \quad (3.69)$$

D'où :

$$\begin{bmatrix} S_{11} & 0 \\ S_{21} & S_{22} \end{bmatrix} = \begin{bmatrix} -\text{sign}(A) & 0 \\ -X\text{sign}(A) + \text{sign}(B)X & -\text{sign}(B) \end{bmatrix} \quad (3.70)$$

Finalement, $S_{21} = -X\text{sign}(A) + \text{sign}(B)X$. Cette nouvelle équation de Sylvester peut être réduite à $S_{21} = -2X$ dans le cas d'une équation de Sylvester stable (voir section précédente pour la relation avec les équations de Lyapunov). Malheureusement, cette méthode ne peut être appliquée au cas général (3.66). Une voie est ouverte dans le chapitre 3 pour contourner ce problème.

2.4.5 Liens avec les LMI

Plusieurs séries de tests ont été effectuées en vue de comparer l'approche LMI et l'approche itérative du calcul de la fonction signe matricielle pour résoudre des équations de Riccati linéaires de la forme (3.51). Ces tests ont été effectués pour des précisions identiques et sur des systèmes de différentes tailles. Les résultats ont montré que quelque soit la méthode utilisée pour calculer la solution de Riccati par la fonction signe, les solutions trouvées via les LMI prennent de manière significative plus de temps de calcul, c'est ce que nous montrons dans le tableau suivant :

Précision $\varepsilon = 10^{-8}$	LMI	Itérations de Newton	Itérations de Newton-Shultz
$n = 2$	0.042	0.0007	0.0011
$n = 5$	0.044	0.0015	0.0022
$n = 10$	0.055	0.0029	0.0046
$n = 20$	0.242	0.0104	0.0191
$n = 40$	12.225	0.041	0.067
$n = 50$	53.657	0.066	0.094
$n = 200$	-	3.242	5.405

Table 2.4: Comparaison du temps de calcul entre LMI et la fonction signe matricielle

Dans ce tableau, n correspond à la taille de la matrice A du système. Chaque système est choisi aléatoirement à l'aide de la fonction 'rss' de Matlab qui génère des modèles continus aléatoires. Pour chaque taille de matrices, on montre une moyenne du temps de calcul nécessaire à la résolution d'une équation de Riccati sur un ensemble de 20 systèmes différents. Afin de ne pas comparer à nouveau toutes les méthodes de calcul de la fonction signe matricielle, nous présentons ici la méthode classique de Newton et les itérations de Newton-Shultz, qui sont essentiellement utilisées lorsque l'on traitera le cas paramétré. On observe donc dans le tableau 2.4 que le temps de calcul nécessaire à la résolution d'une équation de Riccati de la forme (3.51) est toujours plus élevé pour l'utilisation des LMI qu'avec les 2 méthodes utilisant la fonction signe matricielle. De plus, en fonction de l'évolution de la taille des systèmes étudiés, on voit clairement que la méthode des LMI

croît fortement en regard des méthodes de Newton et de Newton-Shultz. On peut conclure de ce tableau que l'utilisation de la fonction signe pour la résolution d'équations de Riccati linéaire sur des systèmes de grande taille est préférable en terme de temps de calcul, pour une précision similaire.

Remarque 2.1 : Dans cet exemple, la LMI minimise la trace de la matrice de Lyapunov sous contrainte d'inégalité de Riccati.

2.5 Conclusion

Ce paragraphe marque la fin de ce chapitre dédié à la fonction signe matricielle. L'idée était de présenter les bases nécessaires à la compréhension des chapitres suivants. Les chapitres à suivre sont souvent basés sur les publications faites au cours de la thèse. Ils présentent les résultats obtenus pour la résolution de divers problèmes ouverts dans l'automatique. Il s'agira essentiellement de résoudre des équations de Lyapunov, de Riccati et de Sylvester, dont le lien vient d'être montré dans cette section. Cette fois, ces équations seront dépendantes d'un vecteur de paramètres et avec plusieurs types de dépendances paramétriques. Le chapitre qui suit sera consacré à la résolution d'équations de Riccati et de Lyapunov utilisant d'une part des méthodes « directes » et d'autre part l'écriture intégrale de la fonction signe matricielle. Dans les chapitres suivants, des classes de matrices $Z(\theta)$ seront définies pour chaque méthode. Ces classes varient en fonction de certaines contraintes ou des différents types de paramétrisation : monomiale, polynomiale, rationnelle...

Chapitre 3 :

Méthodes non-itératives

*La maturité de l'homme est d'avoir retrouvé le sérieux
qu'on avait au jeu quand on était enfant.*

Alain Damasio

Dans ce chapitre nous développerons trois points importants dans le sens de la résolution d'équations de Lyapunov et de Riccati. Premièrement, nous introduirons ce qu'on appelle la méthode directe de résolution d'une équation de Lyapunov. Cette partie se base sur la décomposition de matrices. La méthode présentée dans cette section ne s'appuie pas sur le calcul de la fonction signe matricielle. Cette première partie aura pour but de résoudre des problèmes d'analyse H_2 robuste et de problèmes de commande H_2 structurée, pour des systèmes LTI multiparamétrés. Ce choix a été fait dans le sens où ce sont des problèmes très ouverts encore dans la littérature. Ces problèmes ont été ramenés dans cette partie à un problème de résolution d'équations de Lyapunov paramétrée. La deuxième partie en revanche, concerne l'écriture intégrale de la fonction signe matricielle. En effet cette méthode est également directe et ne nécessite aucune itération. Toujours dans l'idée de résoudre des problèmes d'analyse H_2 robuste et de problèmes de commande H_2 structurée nous nous ramenons ici à des résolutions d'équation de Riccati et de Lyapunov paramétrées. Enfin, la dernière partie aura pour but d'ouvrir une brèche dans l'affranchissement du calcul de l'intégrale pour des matrices d'ordre 2.

3.1 Méthode directe de résolution d'équations de Lyapunov

Ce chapitre présente une nouvelle méthode pour traiter de l'analyse H_2 robuste et des problèmes de commande H_2 structurée pour des systèmes multi paramétrés linéaires à temps invariants (LTI). Ces deux problèmes sont encore ouverts et seront formulés, dans cette partie, comme problèmes de résolution exacte d'équation de Lyapunov . Deux résultats sont proposés, une méthode d'inversion directe sur des matrices particulières puis une transformation de Fourier discrète (DFT) pour le calcul d'inverses de matrices polynomiales.

Le critère de performance H_2 joue un rôle important dans l'analyse de performance et la commande des systèmes linéaires. Le problème de contrôle H_2 est connu pour être un des premiers problèmes de commande optimale résolu de manière analytique [AM90]. Depuis

les années 1970, de nombreux travaux ont trait à la recherche des limites de stabilité pour le contrôleur H_2 -optimal. Dans ce sens, des chercheurs ont proposé une analyse H_2 robuste en une combinaison de robustesse et de performance H_2 (voir [PF00] et références incluses). On trouve beaucoup de travaux dans la littérature concernant la performance H_2 robuste [Fer 97], [Sto93] et des problèmes liés aux différentes combinaisons de performances H_2 et H_∞ [ZGB+94], [DZG+94]. On notera par exemple que quelques caractérisations pour la performance H_2 robuste ont été proposées dans le domaine fréquentiel se basant sur la stabilité robuste et la performance H_∞ [Pag99a], [Pag99b]. Bien que l'évaluation de la performance H_2 robuste soit un sujet très abondant dans la littérature, c'est un domaine toujours ouvert.

De plus, un important problème de contrôle survient quand une structure spécifique d'un modèle général de commande est considérée. Cette structure dépend souvent de la structure du système elle-même, divisée en sous-systèmes. Elle dépend aussi des signaux accessibles pour la mesure et de la liberté de manipuler les variables de chaque contrôleur inclus dans le régulateur global. Trouver un régulateur H_2 sous contrainte de structure est un problème d'optimisation difficile. Excepté dans les cas où la structure du contrôleur satisfait la propriété d'invariance quadratique [RL02], [Vou01], le problème est généralement non-convexe. Même si beaucoup de procédures numériques sous-optimales [YC05] ont été développées afin de réduire le conservatisme et le temps de calcul pour ces problèmes, trouver des méthodes efficaces reste un problème difficile.

Cette partie propose une nouvelle méthode pour traiter d'analyse H_2 robuste et de problèmes de commande H_2 structurée pour des systèmes LTI multiparamétrés. Ces deux problèmes, introduits dans les précédents paragraphes, seront remaniés sous la forme de problèmes visant à trouver la solution exacte d'une équation de Lyapunov paramétrique.

3.1.1 Définition du problème

3.1.1.1 Norme H_2 d'un système

Rappelons d'abord la définition de la norme H_2 d'un système LTI. Considérons le système sous forme d'état suivant :

$$(\Sigma) \quad \begin{cases} \dot{x} = Ax + B_w w + Bu \\ z = Cx + Du \end{cases}, x(0) = 0 \quad (3.1)$$

où $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, $w \in \mathbb{R}^{n_w}$ et $z \in \mathbb{R}^{n_z}$. On notera $H(s)$ la fonction de transfert correspondante (pour $u = 0$). La norme H_2 du système (3.1) est définie comme suit :

$$\|H\|_2^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \text{Tr} \left(H(j\omega)^* H(j\omega) \right) d\omega \stackrel{\Delta}{=} \gamma_2^2 \quad (3.2)$$

Ce qui peut être également écrit :

$$\|H\|_2^2 = \text{Tr} \left(B_w^T \left(\int_0^{+\infty} e^{A^T t} C^T C e^{A t} dt \right) B_w \right) = \text{Tr} (B_w^T X B_w) \quad (3.3)$$

où X est le gramien d'observabilité du système (3.1). Le calcul de X peut être fait en résolvant l'équation de Lyapunov suivante (voir [Ham82] pour des solutions numériques dans le cas LTI) :

$$A^T X + X A + C^T C = 0 \quad (3.4)$$

3.1.1.2 Analyse H_2 robuste

Considérons maintenant la représentation d'état du système suivant :

$$(\Sigma_\theta) \quad \begin{cases} \dot{x} = A(\theta)x + B_w(\theta)w \\ z = C(\theta)x \end{cases} \quad (3.5)$$

où $x \in \mathbb{R}^{n_x}$, $w \in \mathbb{R}^{n_w}$ et $z \in \mathbb{R}^{n_z}$. $\theta = (\theta_i)_{i \in \{1, \dots, p\}}$, $\theta_i \in \mathbb{R}$ représentent les bornes des incertitudes paramétrées présentes dans (3.5). Les matrices $A(\theta)$, $B_w(\theta)$ et $C(\theta)$ sont considérées comme étant à dépendance rationnelle en les paramètres θ_i .

Sans perte de généralité, θ est supposé appartenir à la boule unitaire B_θ pour la norme-2 induite :

$$H(s, \theta) = C(\theta)(sI - A(\theta))^{-1} B_w(\theta) \quad (3.6)$$

$H(s, \theta)$ étant la fonction de transfert correspondante. Ayant (3.6), la norme H_2 robuste du système (3.5) est définie ici par :

$$\begin{aligned} \sup_{\theta \in B_\theta} \|H(s, \theta)\|_2^2 &= \sup_{\theta \in B_\theta} \frac{1}{2\pi} \int_{-\infty}^{+\infty} \text{Tr} \left(H(j\omega, \theta)^* H(j\omega, \theta) \right) d\omega \\ \sup_{\theta \in B_\theta} \|H(s, \theta)\|_2^2 &\stackrel{\Delta}{=} \gamma_{2, \text{robust}}^2 \end{aligned} \quad (3.7)$$

Considérons :

$$\gamma_{2, \text{robust}}^2 = \sup_{\theta \in B_\theta} \text{Tr} \left(B_w(\theta)^T X(\theta) B_w(\theta) \right) \quad (3.8)$$

où $X(\theta)$ peut être calculé en résolvant l'équation de Lyapunov suivante :

$$A(\theta)^T X(\theta) + X(\theta) A(\theta) + C(\theta)^T C(\theta) = 0 \quad (3.9)$$

Nous pouvons à présent introduire le premier problème considéré dans cette partie.

Problème 3.1 : Sous l'hypothèse de stabilité robuste, trouver la pire norme H_2 $\gamma_{2,robust}$ du système (Σ_θ) , (3.5), et définie par (3.7).

3.1.1.3 Contrôle H_2 par retour d'état structuré

Dans cette section, nous considérons le système LTI (Σ) avec la représentation d'état (3.1). Définissons K un retour d'état statique tel que $u = -Kx$ et T_{zw} la matrice de transfert en boucle fermée donnée par :

$$T_{zw} \triangleq \left[\begin{array}{c|c} A - BK & B_w \\ \hline C - DK & 0 \end{array} \right] \quad (3.10)$$

La contrainte de structure sur le contrôleur est définie par :

$$K = \Lambda_K \times K, \quad \Lambda_K \in \{0, 1\}^{n_u \times n_x} \quad (3.11)$$

où $\Lambda_K \in \{0, 1\}^{n_u \times n_x}$ signifie des matrices réelles de dimension $n_u \times n_x$ contenant seulement soit 0 soit 1.

Problème 3.2 : Le problème de retour d'état structuré H_2 optimal consiste à trouver le contrôleur optimal K_{opt} , sous contrainte de structure (3.11) tel que :

$$K_{opt} = \arg \min_K \|T_{zw}\|_2 \quad (3.12)$$

3.1.2 Résolution d'équations de Lyapunov paramétrées

3.1.2.1 Méthode d'inversion directe

Considérons la classe de matrices polynomiales multi-paramétrées de la forme :

$$Z(\theta) \triangleq \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n} \quad (3.13)$$

avec $\bar{Z} \in \mathbb{R}^{n \times nm^p}$, $\bar{\theta} = [\bar{\theta}_1 \otimes \dots \otimes \bar{\theta}_p]^T \in \mathbb{R}^{m^p}$ et $\bar{\theta}_k = [\theta_k^0 \ \theta_k^1 \ \dots \ \theta_k^{m-1}]^T \in \mathbb{R}^m$, $k \in \{1, \dots, p\}$. Nous définissons également la matrice de permutation $T \in \{0, 1\}^{nm^p \times nm^p}$ telle que $(I_n \otimes \bar{\theta}) = T(\bar{\theta} \otimes I_n)$.

Remarque 3.1 : Pour la suite, $Z(\theta)$ est supposée non singulière, soit $\det(Z(\theta)) \neq 0$.

Une arithmétique basique sur la classe de matrices paramétrées introduite ci-dessus est présentée dans les lemmes suivants.

Lemme 3.1 : Posons $Z(\theta) = \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ et $V(\theta) = \bar{V}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ avec $\bar{\theta} = [\bar{\theta}_1 \otimes \dots \otimes \bar{\theta}_p]^T$, alors la somme $Z(\theta) + V(\theta)$, le produit $Z(\theta)V(\theta)$ et le produit de Kronecker $Z(\theta) \otimes V(\theta)$ peuvent être calculés comme suit :

$$Z(\theta) + V(\theta) = (\bar{Z} + \bar{V})(\bar{\theta} \otimes I_n) \quad (3.14)$$

$$Z(\theta) \cdot V(\theta) = \bar{Z}(I_{m^p} \otimes \bar{V})((\bar{\theta} \otimes \bar{\theta}) \otimes I_n) \quad (3.15)$$

$$Z(\theta) \otimes V(\theta) = (\bar{Z} \otimes \bar{V})(I_n \otimes T \otimes I_n)((\bar{\theta} \otimes \bar{\theta}) \otimes I_{n^2}) \quad (3.16)$$

On note une partition de \bar{Z} donnée par $\bar{Z} = [Z_1 \ \dots \ Z_{m^p}]$, $Z_{i \in \{1, \dots, m^p\}} \in \mathbb{R}^{n \times n}$:

$$\text{Tr}(Z(\theta)) = [\text{Tr}(Z_1) \ \dots \ \text{Tr}(Z_{m^p})] \bar{\theta} \quad (3.17)$$

$$(Z(\theta))^T = \bar{Z}'(\bar{\theta} \otimes I_n) = [Z_1^T \ \dots \ Z_{m^p}^T](\bar{\theta} \otimes I_n) \quad (3.18)$$

Les preuves de ce lemme sont directes et donc non décrites ici.

Considérons maintenant l'équation de Lyapunov paramétrique suivante :

$$A(\theta)^T X(\theta) + X(\theta) A(\theta) + Q(\theta) = 0 \quad (3.19)$$

où $A(\theta) = \bar{A}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ et $Q(\theta) = \bar{Q}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ sont des matrices paramétrées de la forme (3.13).

Théorème 3.1 : La solution de l'équation de Lyapunov paramétrique (3.19) est donnée par :

$$\text{vec}(X(\theta)) = \left[\sum_{k=1}^{n^2} \frac{p_{k-1}}{p_{n^2}} \left(\sum_{i=0}^{n^2-k} Z(\theta)^i \right) \right] \text{vec}(Q(\theta)) \quad (3.20)$$

$$\text{où } \begin{cases} p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} \left(\sum_{i=0}^j \text{Tr}(Z(\theta)^i) \right) \\ p_0 = 1 \end{cases} \text{ et } Z(\theta) = [(I_n \otimes \bar{A}')T + (\bar{A}' \otimes I_n)](\bar{\theta} \otimes I_{n^2}).$$

Preuve : (3.19) peut être réécrite comme suit :

$$(I_n \otimes A(\theta)^T + A(\theta)^T \otimes I_n) \text{vec}(X(\theta)) = -\text{vec}(Q(\theta))$$

Notons $Z(\theta) = (I_n \otimes A(\theta)^T + A(\theta)^T \otimes I_n)$. L'équation amène à :

$$Z(\theta) = (I_n \otimes \bar{A}'(\bar{\theta} \otimes I_n) + \bar{A}'(\bar{\theta} \otimes I_n) \otimes I_n) = [(I_n \otimes \bar{A}')T + (\bar{A}' \otimes I_n)](\bar{\theta} \otimes I_{n^2})$$

où \bar{A}' est définie comme en (3.18). Se basant sur le théorème de Cayley-Hamilton nous avons :

$$Z^{-1} = -\frac{1}{p_{n^2}} [Z^{n^2-1} + p_1 Z^{n^2-2} + \dots + p_{n^2-2} Z + p_{n^2-1} I_{n^2}]$$

avec : $p_k = -\frac{1}{k}(s_k + p_1 s_{k-1} + \dots + p_{k-1} s_1)$, $s_j = \text{Tr}(Z^j)$, $j = [1, k]$ et $p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} s_j$, $p_0 = 1$.

L'expression de Z^{-1} est une somme finie de puissance de Z dépendante de sa dimension. Les puissances $n^2 - k$ de Z sont données par :

$$Z^{n^2-k} = \sum_{i=0}^{n^2-k} Z(\theta)^i \quad (3.21)$$

Finalement Z^{-1} s'écrit :

$$Z^{-1} = \sum_{k=1}^{n^2} -\frac{p_{k-1}}{p_{n^2}} Z^{n^2-k} = \sum_{k=1}^{n^2} -\frac{p_{k-1}}{p_{n^2}} \left(\sum_{i=0}^{n^2-k} Z(\theta)^i \right) \quad (3.22)$$

facilement calculable à l'aide du Lemme 3.1. ■

Le résultat du Théorème 3.1 est applicable pour le cas de l'équation de Lyapunov généralisée. Son expression est la suivante :

$$A(\theta)^T X(\theta) E(\theta) + E(\theta)^T X(\theta) A(\theta) + Q(\theta) = 0 \quad (3.23)$$

où $E(\theta) = \bar{E}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$.

Corollaire 3.1: La solution de l'équation de Lyapunov généralisée (3.23) est donnée par:

$$\text{vec}(X(\theta)) = \left[\sum_{k=1}^{n^2} \frac{p_{k-1}}{p_{n^2}} \left(\sum_{i=0}^{n^2-k} Z(\theta)^i \right) \right] \text{vec}(Q(\theta)) \quad (3.24)$$

$$\text{où } \begin{cases} p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} \left(\sum_{i=0}^j \text{Tr}(Z(\theta)^i) \right) \\ p_0 = 1 \end{cases} \text{ et } \begin{cases} Z(\theta) = ((\bar{E}' \otimes \bar{A}') + (\bar{A}' \otimes \bar{E}')) \tilde{T} ((\bar{\theta} \otimes \bar{\theta}) \otimes I_{n^2}) \\ \tilde{T} = I_n \otimes T \otimes I_n \end{cases}$$

Preuve : (3.23) peut être réécrit de la manière suivante :

$$\left(E(\theta)^T \otimes A(\theta)^T + A(\theta)^T \otimes E(\theta)^T\right) \text{vec}(X(\theta)) = -\text{vec}(Q(\theta))$$

Notons : $Z(\theta) = \left(E(\theta)^T \otimes A(\theta)^T + A(\theta)^T \otimes E(\theta)^T\right)$. Se basant sur le Lemme 3.1, l'équation se réécrit comme suit:

$$\begin{cases} Z(\theta) = \left((\bar{E}' \otimes \bar{A}') + (\bar{A}' \otimes \bar{E}')\right) \tilde{T} \left((\bar{\theta} \otimes \bar{\theta}) \otimes I_{n^2}\right) \\ \tilde{T} = I_n \otimes T \otimes I_n \end{cases}$$

La fin de la preuve suit directement celle du Théorème 3.1. ■

Remarque 3.2 : Le Corollaire 3.1 est important pour la généralisation au cas multi-paramétrique à dépendance rationnelle de l'équation générale de Lyapunov associée au système (Σ_θ) donné par (3.5).

Exemple 3.1 : considérons un système paramétré admettant la représentation d'état suivante :

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{bmatrix} 1+\theta_1 & 0 \\ 2+\theta_2 & \frac{1}{\theta_2+3} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad \theta_i \in [-1 \quad 1]_{i=1,2}$$

Cette représentation d'état peut être réécrite sous la forme :

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{E(\theta)} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \zeta \end{pmatrix} = \underbrace{\begin{bmatrix} 1+\theta_1 & 0 & 0 \\ 2+\theta_2 & 0 & 1 \\ 0 & -1 & \theta_2+3 \end{bmatrix}}_{A(\theta)} \begin{pmatrix} x_1 \\ x_2 \\ \zeta \end{pmatrix}, \quad \theta_i \in [-1 \quad 1]_{i=1,2}$$

Reprenant la définition (3.13) nous posons $A(\theta) = \bar{A}(\bar{\theta} \otimes I_3)$ et $E(\theta) = \bar{E}(\bar{\theta} \otimes I_3)$:

$$\bar{\theta} = \begin{pmatrix} 1 \\ \theta_1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \theta_2 \end{pmatrix}, \quad \bar{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{ et}$$

$$\bar{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Ensuite, la solution de (3.23) peut être trouvée grâce au résultat (3.24) en utilisant les opérations définies dans le Lemme 3.1.

3.1.2.2 Méthode de transformée de Fourier Discrète

Nous rappellerons dans un premier temps la définition d'une paire de matrice DFT.

Définition 3.1 : Les deux fonctions de matrice $F(l)$ et $G(r)$, $l, r = 0, 1, \dots, N-1$ sont dites paires DFT si les équations suivantes sont vérifiées :

$$F(l) = \bar{G}(\rho_l \otimes I_n) \in \mathbb{C}^{n \times n}$$

$$G(r) = (N^{-1}) \bar{F}(\delta_r \otimes I_n) \in \mathbb{C}^{n \times n}$$

$$\text{avec :} \quad \bar{F} = [F(0) \quad \dots \quad F(N-1)], \quad \bar{G} = [G(0) \quad \dots \quad G(N-1)],$$

$$\rho_l = [\rho^0 \quad \rho^l \quad \dots \quad \rho^{l(N-1)}]^T, \quad \delta_r = [\delta^0 \quad \delta^r \quad \dots \quad \delta^{r(N-1)}]^T, \quad \rho = e^{\left(\frac{-2\pi j}{N}\right)} \text{ et } \delta = \rho^{-1}.$$

Une extension de la Définition 3.1 au cas de plusieurs variables discrètes $l_k, r_k = 0, 1, \dots, N_k - 1$ ($k = 1, 2, \dots, p$) est donnée ci-après.

Définition 3.2 : Les deux fonctions de matrice $F(l_1, \dots, l_p)$ et $G(r_1, \dots, r_p)$ sont dites paires DFT si les équations suivantes sont vérifiées :

$$F(l_1, \dots, l_p) = \bar{G}(\bar{\rho} \otimes I_n) \in \mathbb{C}^{n \times n}$$

$$G(r_1, \dots, r_p) = \left(\prod_{k=1}^p N_k^{-1} \right) \bar{F}(\bar{\delta} \otimes I_n) \in \mathbb{C}^{n \times n}$$

$$\text{avec :} \quad \bar{F} = [F(l_1, \dots, l_p)]_{l_k=[0, N_k-1] \ (k \in [1, p])}, \quad \bar{\rho} = [\bar{\rho}_1 \otimes \dots \otimes \bar{\rho}_p]^T,$$

$$\bar{\rho}_k = [\rho_k^0 \quad \rho_k^{l_k} \quad \dots \quad \rho_k^{l_k(N_k-1)}]^T, \quad \rho_k = e^{\left(\frac{-2\pi j}{N_k}\right)}, \quad \delta_k = e^{\left(\frac{2\pi j}{N_k}\right)}, \quad \bar{G} = [G(r_1, \dots, r_p)]_{r_k=[0, N_k-1] \ (k \in [1, p])} \text{ et}$$

$$\bar{\delta}_k = [\delta_k^0 \quad \delta_k^{r_k} \quad \dots \quad \delta_k^{r_k(N_k-1)}]^T.$$

Se basant sur les travaux [Yeu84] et [YK88], nous proposons dans cette section d'utiliser une méthode DFT pour le calcul de l'inverse exacte d'une matrice à dépendance polynomiale (sous l'hypothèse de non singularité, voir Remarque 3.1).

Dans ce but, considérons la matrice $Z(\theta) = \bar{Z}(\bar{\theta} \otimes I_n)$ de la forme (3.13). Par définition :

$$Z(\theta)^{-1} = \frac{\text{adj}(Z(\theta))}{\det(Z(\theta))} \quad (3.25)$$

où la matrice $\text{adj}(Z(\theta))$ peut être réécrite :

$$\text{adj}(Z(\theta)) = Z_a(\theta) = \bar{Z}_a(\bar{\theta} \otimes I_n) \quad (3.26)$$

Le but ici est de calculer \bar{Z}_a . Définissons le changement de variable suivant :

$$\theta_k = \rho_k^{l_k}, \quad k=1,2,\dots,p \quad (3.27)$$

En combinant (3.26) et (3.27) nous obtenons:

$$Z_a(\rho_1^{l_1}, \rho_2^{l_2}, \dots, \rho_p^{l_p}) = Z_\rho(l_1, l_2, \dots, l_p) \quad (3.28)$$

Puis, en calculant numériquement la fonction $Z_\rho(l_1, l_2, \dots, l_p)$ et en prenant son inverse DFT :

$$\bar{Z}_a(r_1, \dots, r_p) = \left(\prod_{k=1}^p N_k^{-1} \right) \bar{Z}_\rho(\bar{\delta} \otimes I_n) \quad (3.29)$$

On obtient finalement la matrice \bar{Z}_a voulue. La même procédure est appliquée pour le calcul du déterminant $\det(Z(\theta))$.

Exemple 3.2 : Considérons la matrice :

$$Z(\theta) = \begin{bmatrix} \theta_1 \theta_2 + 3 & -7\theta_1 \\ \theta_1 - \theta_2 & \theta_2 \end{bmatrix} = \bar{Z}(\bar{\theta} \otimes I_2)$$

avec $\bar{Z} = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & -7 & 1 & 0 \\ 0 & 0 & -1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$ et $\bar{\theta} = \begin{pmatrix} 1 \\ \theta_1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \theta_2 \end{pmatrix}$.

Pour cet exemple nous avons : $N_1 = 2$, $N_2 = 2$, $l_1 \in \{0,1\}$, $l_2 \in \{0,1\}$, $\theta_k = \rho_k^{l_k}$, $k=1,2$,

$\rho_1 = e^{\frac{2\pi j}{N_1}}$, $\rho_2 = e^{\frac{2\pi j}{N_2}}$ et $\bar{\rho} = \begin{pmatrix} 1 \\ \rho_1^{l_1} \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \rho_2^{l_2} \end{pmatrix}$. On obtient ensuite par simple calcul :

$$\bar{Z}_\rho = \begin{bmatrix} 1 & 7 & -1 & 7 & 1 & -7 & -1 & -7 \\ 0 & 4 & -2 & 2 & 2 & 2 & 0 & 4 \end{bmatrix}$$

L'inverse DFT de cette matrice nous donne :

$$\bar{Z}_a = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 7 & 0 & 0 \\ 0 & 3 & 1 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}$$

Ce qui nous mène à l'expression suivante :

$$\text{adj}(Z(\theta)) = \bar{Z}_a (\bar{\theta} \otimes I_2) = \begin{bmatrix} \theta_2 & 7\theta_1 \\ \theta_2 - \theta_1 & \theta_1\theta_2 + 3 \end{bmatrix}$$

3.1.2.3 Exemples numériques

Exemple 3.3 : Considérons le système paramétré (Σ_θ) défini par (3.5) avec les valeurs suivantes :

$$A(\theta) = \begin{bmatrix} \theta_1 & \theta_2 \\ -1 & 2\theta_2 \end{bmatrix}, B_w = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, C = [1 \quad -2], \theta = \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} \in \Gamma_\theta.$$

On notera également $Q = C^T C$. Considérant la stabilité robuste pour $\theta \in \Gamma_\theta$, nous utilisons ici le Théorème 3.1 dans le but de calculer la norme H_2 (3.7) de (Σ_θ) . L'objectif est de trouver la solution de l'équation de Lyapunov paramétrique :

$$A(\theta)^T X(\theta) + X(\theta) A(\theta) + Q = 0$$

Nous avons dans ce cas :

$$\bar{A}' = \begin{bmatrix} 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 \end{bmatrix}$$

et $\bar{\theta} = \begin{pmatrix} 1 \\ \theta_1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \theta_2 \end{pmatrix} = (1 \quad \theta_2 \quad \theta_1 \quad \theta_1\theta_2)^T$. En utilisant la formulation de $Z(\theta)$ associée au

Théorème 3.1, nous avons :

$$\bar{Z} = \begin{bmatrix} 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Le théorème 3.1 nous donne finalement l'expression de $X(\theta)$:

$$X(\theta) = \begin{bmatrix} \frac{-7\theta_2 + 2\theta_1\theta_2 + 4\theta_2^2 + 4}{2\theta_2(2\theta_1 + 1)(\theta_1 + 2\theta_2)} & \frac{\theta_1(4\theta_2 - 2) + \theta_2^2}{\theta_2(2\theta_1 + 1)(\theta_1 + 2\theta_2)} \\ \frac{\theta_1(4\theta_2 - 2) + \theta_2^2}{\theta_2(2\theta_1 + 1)(\theta_1 + 2\theta_2)} & -\frac{1}{\theta_2} - \frac{\theta_1(2\theta_2 - 1) + (\theta_2^2/2)}{\theta_2(2\theta_1 + 1)(\theta_1 + 2\theta_2)} \end{bmatrix}$$

La seconde méthode, utilisant l'application de la méthode DFT présentée précédemment donne :

$$Z(\theta) = \begin{bmatrix} 2\theta_1 & -1 & -1 & 0 \\ \theta_2 & \theta_1 + 2\theta_2 & 0 & -1 \\ \theta_2 & 0 & \theta_1 + 2\theta_2 & -1 \\ 0 & \theta_2 & \theta_2 & 4\theta_2 \end{bmatrix}$$

et :

$$adj Z(\theta) = (\theta_1 + 2\theta_2) \begin{bmatrix} 2\theta_2(2\theta_1 + 4\theta_2 + 1) & 4\theta_2 & 4\theta_2 & 2 \\ -4\theta_2^2 & 2\theta_2(4\theta_1 + 1) & -2\theta_2 & 2\theta_1 \\ -4\theta_2^2 & -2\theta_2 & 2\theta_2(4\theta_1 + 1) & 2\theta_1 \\ 2\theta_2^2 & -2\theta_1\theta_2 & -2\theta_1\theta_2 & 2(\theta_1^2 + 2\theta_1\theta_2 + \theta_2^2) \end{bmatrix}$$

La même procédure est appliquée pour le calcul du déterminant $\det(Z(\theta))$, ce qui nous donne finalement:

$$Z(\theta)^{-1} = \left(\frac{1}{8\theta_1\theta_2 + 3\theta_1^2\theta_2^2 + 4\theta_1^2\theta_2 + 32\theta_1\theta_2^3 + 16\theta_1\theta_2^2 + 16\theta_2^3} \right) adj Z(\theta)$$

On pourra vérifier simplement qu'en multipliant la dernière matrice par $-vec(Q)$ on obtient la même matrice $X(\theta)$ trouvée précédemment à l'aide du Théorème 3.1.

Ayant obtenu la solution explicite de la solution paramétrique $X(\theta)$, nous pouvons calculer le gradient du critère suivant : $J(X(\theta)) = Tr(B_w(\theta)^T X(\theta) B_w(\theta))$.

La résolution de l'équation $\frac{\partial J(X(\theta))}{\partial \theta} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ donne la paire de solutions suivante (Fig. 3.1) :

$$\begin{cases} \theta_1 = -0.317 \\ \theta_2 = 0.085 \end{cases}$$

La valeur de la norme H_2 robuste est : $\gamma_{2,robust} = 25.483$.

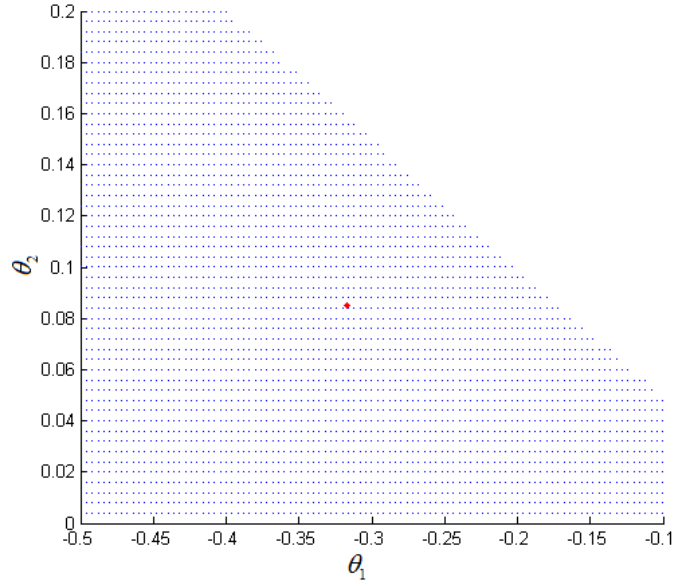


Fig. 3.1 : Performance H_2 robuste.

La figure 3.1 montre l'ensemble des régulateurs (en bleu) possibles dans les intervalles θ_1 et θ_2 . Le point rouge correspond au point calculé précédemment.

Exemple 3.4 : Synthèse H_2 structurée

Considérons le système (Σ) sous la forme d'état donnée par (3.1) avec les valeurs suivantes :

$$\begin{cases} A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}, & x_0 = B_w = \begin{pmatrix} -1 & 2 \end{pmatrix}^T & B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, & D = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{cases}$$

Ici, x_0 est une condition initiale donnée. Le but est de trouver le retour d'état H_2 décentralisé K tel que $u = -Kx$ et $K = \begin{bmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{bmatrix}$. L'objectif est ensuite de résoudre l'équation de Lyapunov paramétrique suivante :

$$(A - BK(\theta))^T X(\theta) + X(\theta)(A - BK(\theta)) + Q(\theta) = 0$$

où $Q(\theta) = K(\theta)^T RK(\theta) - C^T DK(\theta) - K(\theta)^T D^T C$ et $R = (D^T D) = I_2$.

Dans ce cas, nous avons : $\bar{\theta} = (1 \ \theta_1)^T \otimes (1 \ \theta_2)^T$ et :

$$(A - BK(\theta))^T = \begin{pmatrix} 1 & 2 & 0 & 0 & -1 & 0 & 0 & 0 \\ 2 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \end{pmatrix} (\bar{\theta} \otimes I_2)$$

En utilisant la formulation de $Z(\theta)$ associée au Théorème 3.1, on a :

$$\bar{Z} = \begin{pmatrix} 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 2 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 2 & 2 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 2 & 2 & 0 & 1 & 1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Finalement, le résultat du Théorème 3.1 nous donne l'expression de $X(\theta)$ suivante :

$$X(\theta) = \frac{1}{\theta_1 + \theta_2 - \theta_1 \theta_2 + 3} \begin{bmatrix} \theta_1 - 2\theta_2 + \theta_1 \theta_2 + \frac{\theta_1^2 - \theta_1^2 \theta_2}{2} & \theta_1 + \theta_2 - \theta_1 \theta_2 \\ \theta_1 + \theta_2 - \theta_1 \theta_2 & \theta_2 - 2\theta_1 + \theta_1 \theta_2 + \frac{\theta_2^2 - \theta_1 \theta_2^2}{2} \end{bmatrix}$$

Le même résultat est obtenu en utilisant la méthode DFT. La résolution de l'équation

$\frac{\partial J(X(\theta))}{\partial \theta} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ donne quatre paires de solutions. L'unique paire de solution stabilisante

est donnée par : $\begin{cases} \theta_1 = 3.2679 \\ \theta_2 = 3.7321 \end{cases}$. Enfin, la valeur de la norme H_2 est : $J(X(\theta)) = 2.279$.

Remarque 3.3 : Les deux méthodes présentées dans cette partie dans le but d'obtenir la solution $X(\theta)$ d'une équation de Lyapunov paramétrique ont sensiblement le même temps de calcul pour des systèmes d'ordres faibles. Il a été observé, néanmoins, qu'utiliser la méthode d'inversion directe pour des systèmes de dimension plus élevée demande un temps de calcul beaucoup plus grand et entraîne des problèmes de mémoire pour des matrices \bar{Z} de très grande taille. En revanche, comme la méthode DFT n'implique pas d'augmentation de taille de matrices, le temps de calcul est moins affecté par l'ordre des systèmes considérés.

La partie que l'on vient de traiter a fait l'objet d'un papier publié en conférence [GYC13]. L'objectif a été ensuite de se concentrer sur l'étude de la fonction signe matricielle afin de

voir comment celle-ci peut être importante pour la résolution de ce type d'équations paramétrées. Dans un premier temps, nous nous sommes concentrés sur la forme intégrale de la fonction signe matricielle rappelée et développée dans la section suivante.

3.2 Fonction signe matricielle paramétrée : Définition intégrale

Dans cette partie, nous associons la résolution d'équations de Lyapunov et de Riccati à la fonction signe matricielle intégrale. Nous traiterons également le cas de la stabilité robuste. Deux points importants sont donc repris ici afin de répondre à la problématique. Dans un premier temps, l'extension de l'écriture intégrale de la fonction signe matricielle au cas paramétré et dans un second temps, une représentation particulière des matrices dépendantes de paramètres. Contrairement aux autres parties de cette thèse, celle-ci offre une réponse exacte pour la solution des équations de Lyapunov et de Riccati.

3.2.1 Introduction

L'analyse robuste des systèmes linéaires à temps invariant (LTI) incertains est un des problèmes les plus étudiés dans l'automatique [Bar94], [Bli04]. Une des stratégies pour l'analyse de la stabilité robuste en la reformulation sous forme de problèmes de faisabilité sous contraintes d'inégalités matricielles linéaires (LMI), via le paradigme de Lyapunov [EPA+05], [GPT+07], [Sch06], [TdS01]. De nombreuses contributions dans les approches LMI utilisant des fonctions de Lyapunov sont à noter ([CGT+05], [FAG96], [GAC96], [LP03], [OP07], [PA01] et [SP06]). Quelques conditions non-conservatives et efficaces ont également été obtenues en terme de LMI [Che08], [CGT+09].

L'objectif principal de cette partie est de proposer une alternative pour l'analyse de la stabilité robuste exacte d'une classe de systèmes paramétrés se basant sur la fonction signe matricielle [BHM97], [KL95], [Rob80]. La méthode présentée ici ne s'appuie pas sur la notion de la stabilité au sens de Lyapunov et a l'avantage d'être utilisée pour la solution d'équations de Lyapunov et de Riccati. Cette démarche apparaît comme étant également généralisable au cas multi-paramétrique et donc pouvant traiter des systèmes à temps invariant avec dépendance rationnelle.

Si des algorithmes basés sur la fonction signe matricielle se sont montrés intéressants pour la solution à grande échelle d'équations de Riccati, peu de résultats existent en revanche pour le cas paramétré [RV10]. En fait, quelques travaux sont notables dans la littérature pour des systèmes linéaires invariants à temps continu [KL95], [Q-OH98], [RV09], [KLW90], [KLP92], [KLJ89] et à temps discret [FB99]. Plus précisément, la méthode développée dans [RV10] et [RV09] consiste en l'utilisation des algorithmes itératifs de Newton associés à la fonction signe matricielle et des transformations fractionnelles linéaires (LFT). Bien que prometteuse, cette approche a quelques inconvénients : - Une réduction d'ordre des LFT pour chaque itération de Newton (si cela peut être effectué efficacement au cas d'un seul paramètre, le problème est beaucoup plus complexe au cas

multi-paramétrique). - Le temps de calcul augmente considérablement si l'on désire un résultat précis. Une alternative est proposée, basée sur la définition intégrale de la fonction signe matricielle.

3.2.2 Définitions et méthodes, classe de matrices

Nous présenterons dans un premier temps le résultat au cas mono paramétrique, pour plus de clarté puis l'extension au cas multi-paramétrique.

Dans cette partie, nous considérons une classe de matrices de la forme suivante :

$$Z(\theta) \triangleq \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n} \quad (3.30)$$

avec : $\bar{Z} \in \mathbb{R}^{n \times nw}$ et $\bar{\theta} = \begin{bmatrix} \underbrace{\theta^{-l} \dots \theta^{-1}}_l & \theta^0 & \underbrace{\theta^1 \dots \theta^m}_m \end{bmatrix}^T \in \mathbb{R}^w$, où $w \triangleq l+m+1 \in \mathbb{N}$.

Une autre notation $\bar{\theta} \triangleq \theta^{[-l,m]}$ est également utilisée dans la suite.

La représentation (3.30) de $Z(\theta)$, reprise de [SP06], présente une matrice constante \bar{Z} (qui est une matrice rectangulaire composée de w matrices constantes de taille $n \times n$) et d'un vecteur $\bar{\theta}$ comprenant tous les paramètres de la matrice. Des opérations basiques sur ces classes de matrices sont présentées dans les lemmes suivants.

Lemme 3.2 : Posant $Z(\theta) = \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ et $V(\theta) = \bar{V}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ avec $\bar{\theta} = \theta^{[-l,m]}$ et $\alpha \in \mathbb{R}$ alors les sommes $Z(\theta) + V(\theta)$, $\alpha I_n + Z(\theta)$ et le produit $Z(\theta)V(\theta)$ sont représentés par :

$$Z(\theta) + V(\theta) = (\bar{Z} + \bar{V})(\bar{\theta} \otimes I_n) \quad (3.31)$$

$$Z(\theta)V(\theta) = \bar{Z}(I_w \otimes \bar{V})((\bar{\theta} \otimes \bar{\theta}) \otimes I_n) \quad (3.32)$$

$$\alpha I_n + Z(\theta) = (\alpha \bar{I} + \bar{Z})(\bar{\theta} \otimes I_n) \quad (3.33)$$

avec $\bar{I} = \begin{bmatrix} 0_{n \times l} & I_n & 0_{n \times m} \end{bmatrix} \in \mathbb{R}^{n \times nw}$.

On notera également les puissances de $Z(\theta)$ qui s'écrivent comme suit :

$$Z(\theta)^q = \bar{Z} \left(\prod_{j=1}^{q-1} (I_{w^j} \otimes \bar{Z}) \right) (\bar{\theta}_q \otimes I_n) \quad (3.34)$$

avec $\bar{\theta}_q \triangleq \underbrace{\bar{\theta} \otimes \bar{\theta} \otimes \dots \otimes \bar{\theta}}_q$.

On note une partition de \bar{Z} donnée par $\bar{Z} = [Z_1 \ \cdots \ Z_w]$, $Z_{i \in [1, w]} \in \mathbb{R}^{n \times n}$ alors la trace de $Z(\theta)$ s'écrit comme suit :

$$\text{trace}(Z(\theta)) = [\text{trace}(Z_1) \ \cdots \ \text{trace}(Z_w)] \bar{\theta} \quad (3.35)$$

Les preuves de ces lemmes sont directes et ne sont donc pas présentées ici. Ces résultats montrent que seules des manipulations sur les parties constantes définies sont nécessaires pour effectuer ces opérations basiques. Ce point s'avèrera très utile pour la suite de ce chapitre.

Exemple 3.5 : Considérons la matrice suivante :

$$Z(\theta) \triangleq \begin{bmatrix} -\theta & 1 \\ 0 & -1 \end{bmatrix}, \quad \theta \in \mathbb{R}, \quad \theta \neq 0 \quad (3.36)$$

La représentation de la forme (3.30) pour $Z(\theta)$ est donnée par :

$$Z(\theta) = \begin{bmatrix} 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \end{bmatrix} \otimes I_2 \right).$$

De simples calculs montrent, se basant sur (3.31), montrent que :

$$Z(\theta) + Z(\theta) = \begin{bmatrix} 0 & 2 & -2 & 0 \\ 0 & -2 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \end{bmatrix} \otimes I_2 \right).$$

L'équation (3.33) donne:

$$\alpha \bar{I} + \bar{Z} = \begin{bmatrix} \alpha & 1 & -1 & 0 \\ 0 & \alpha - 1 & 0 & 0 \end{bmatrix}.$$

La trace définie dans (3.35) donne :

$$\text{trace}(Z(\theta)) = \left[\text{trace} \left(\begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix} \right) \quad \text{trace} \left(\begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix} \right) \right] \begin{bmatrix} 1 \\ \theta \end{bmatrix}.$$

Et le carré de la matrice $Z(\theta)$:

$$Z(\theta)^2 = \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \\ \theta \\ \theta^2 \end{bmatrix} \otimes I_2 \right).$$

On remarquera que pour le calcul du carré de $Z(\theta)$, θ est répété, ce qui implique : $\bar{\theta}_q \neq \theta^{[-ql, qm]} \triangleq \bar{\theta}_{\beta(q)}$, avec $\bar{\theta}_{\beta(q)}$ la réalisation minimale où $\beta(q) \triangleq q(m+l)+1$. Dans le but d'éliminer ces doublons, nous définissons la matrice suivante :

$$W_q = [W_1 \quad W_2 \quad \cdots \quad W_w]^T \in \mathbb{R}^{w^q \times \beta}$$

avec :

$$W_1 = \begin{bmatrix} W_{q-1} & 0_{w^{q-1} \times (\beta - w^{q-1})} \end{bmatrix}, W_2 = \begin{bmatrix} 0_{w^{q-1} \times 1} & W_{q-1} & 0_{w^{q-1} \times (\beta - w^{q-1} - 1)} \end{bmatrix}, \dots, W_w = \begin{bmatrix} 0_{w^{q-1} \times (\beta - w^{q-1})} & W_{q-1} \end{bmatrix}$$

Le lemme suivant propose une représentation minimale du calcul de puissances de matrices de la forme (3.30).

Lemme 3.3 : Posons $Z(\theta) = \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ avec $\bar{\theta} = \theta^{[-l, m]}$. La réalisation minimale pour les puissances de $Z(\theta)$ est donnée par :

$$Z(\theta)^q = \bar{Z} \left(\prod_{j=1}^{q-1} (I_{w^j} \otimes \bar{Z}) \right) J_q(\bar{\theta}_{\beta(q)} \otimes I_n) \quad (3.37)$$

avec :

$$J_q \triangleq W_q \otimes I_n \quad (3.38)$$

En termes d'exemple, en reprenant le précédent on trouve :

$$W_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

et la réalisation minimale de $Z(\theta)^2$ est donnée par :

$$Z(\theta)^2 = \begin{bmatrix} 0 & -1 & 0 & -1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \\ \theta^2 \end{bmatrix} \otimes I_2 \right)$$

La présentation du résultat se base sur l'écriture décrite et en utilisant la fonction signe matricielle décrite dans le chapitre précédent. La fonction signe matricielle sera calculée à partir de la définition intégrale de celle-ci.

3.2.3 Extension au cas paramétré

3.2.3.1 Fonction signe matricielle

Le résultat principal, présenté dans le théorème 3.2 se base sur la définition intégrale de la fonction signe matricielle et de la classe de matrices décrite précédemment. L'idée sous-jacente de ce résultat est de reformuler le signe d'une matrice $Z(\theta)$ en une somme finie de puissances de $Z(\theta)$ avec le calcul d'intégrales scalaires de fonctions rationnelles.

Théorème 3.2 : *Considérons une matrice $Z(\theta)$ sous la forme (3.30). Le signe matriciel de $Z(\theta)$, noté $\text{sign}(Z(\theta))$ est donné par :*

$$\text{sign}(Z(\theta)) = \frac{2}{\pi} Z(\theta) \cdot \sum_{k=1}^n \int_0^\infty -\frac{p_{k-1}}{p_n} \left(\sum_{i=0}^{n-k} \binom{n-k}{i} y^{2(n-k-i)} Z(\theta)^{2i} \right) dy \quad (3.39)$$

$$\text{où} \begin{cases} p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} \left(\sum_{i=0}^j \binom{j}{i} y^{2(j-i)} \text{trace}(Z(\theta)^{2i}) \right) \\ p_0 = 1 \end{cases}$$

Preuve : Introduisons tout d'abord $N = y^2 I_n + Z(\theta)^2$, $y \in \mathbb{R}^+$. En se basant sur le théorème de Cayley-Hamilton, nous pouvons noter que :

$$N^{-1} = -\frac{1}{p_n} [N^{n-1} + p_1 N^{n-2} + \dots + p_{n-2} N + p_{n-1} I_n]$$

avec : $p_k = -\frac{1}{k} (s_k + p_1 s_{k-1} + \dots + p_{k-1} s_1)$, $s_j = \text{trace}(N^j)$, $j = [1, k]$ et $p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} s_j$,
 $p_0 = 1$.

L'expression de N^{-1} est une somme finie de puissances de N dépendante de sa taille. Les puissances $n-k$ de N sont notées :

$$N^{n-k} = (y^2 I_n + Z(\theta)^2)^{n-k} = \sum_{i=0}^{n-k} \binom{n-k}{i} y^{2(n-k-i)} Z(\theta)^{2i} \quad (3.40)$$

Et l'inverse N^{-1} :

$$N^{-1} = \sum_{k=1}^n -\frac{p_{k-1}}{p_n} N^{n-k} = \sum_{k=1}^n -\frac{p_{k-1}}{p_n} \left(\sum_{i=0}^{n-k} \binom{n-k}{i} y^{2(n-k-i)} Z(\theta)^{2i} \right) \quad (3.41)$$

Finalement, selon (3.41) nous obtenons directement (3.39). ■

Remarque 3.4 : Le calcul des p_k peut être réduit au calcul de puissances et de traces de matrices de la forme (3.30). On a :

$$p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} s_j, p_0 = 1 \quad (3.42)$$

avec :

$$s_j = \text{trace}(N^j) = \sum_{i=0}^j \binom{j}{i} y^{2(j-i)} \text{trace}(Z(\theta)^{2i}) \quad (3.43)$$

$Z(\theta)^{2i}$ peut être déduit de (3.37) comme suit :

$$(Z(\theta)^{2i}) = \bar{Z} \left(\prod_{j=1}^{2i-1} (I_{w^j} \otimes \bar{Z}) \right) J_{2i} (\bar{\theta}_{\beta(2i)} \otimes I_n) \quad (3.44)$$

et $\text{trace}(Z(\theta)^{2i})$ obtenu directement grâce à (3.35).

Remarque 3.5 : Pour une matrice générale à dépendance paramétrique rationnelle, le calcul de son signe revient à calculer le signe d'une matrice multi-paramétrique. Considérons par exemple la matrice suivante :

$$Z(\theta) = \begin{bmatrix} \frac{1}{\theta+1} & 0 \\ -1 & \theta \end{bmatrix}$$

Elle peut être réécrite sous la forme : $Z(\theta_1, \theta_2) = \begin{bmatrix} \frac{1}{\theta_1} & 0 \\ -1 & \theta_2 \end{bmatrix}$.

C'est également pour cette raison que le résultat proposé sera généralisé au cas multi-paramétrique. Cela nous permet de pouvoir traiter une plus grande classe de matrices, bien que cela engendre un plus grand nombre de paramètres.

Exemple 3.6 : Reprenons la matrice $Z(\theta) \triangleq \begin{bmatrix} -\theta & 1 \\ 0 & -1 \end{bmatrix}$, $\theta \in \mathbb{R}$, $\theta \neq 0$. L'application du théorème 3.2 nous donne :

$$\text{sign}(Z(\theta)) = \frac{2}{\pi} Z(\theta) \left(\int_0^\infty -\frac{1}{p_2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} y^2 I_2 dy + \int_0^\infty -\frac{1}{p_2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} Z(\theta)^2 dy + \int_0^\infty -\frac{p_1}{p_2} I_2 dy \right)$$

avec $p_1 = -(2y^2 + 1 + \theta)$, $p_2 = y^4 + y^2(1 + \theta^2) + \theta^2$. On a ensuite :

$$\text{sign}(Z(\theta)) = \frac{2}{\pi} Z(\theta) \left(\int_0^\infty \frac{y^2 + 1 + \theta^2}{y^4 + y^2(1 + \theta^2) + \theta^2} I_2 dy - \int_0^\infty \frac{1}{y^4 + y^2(1 + \theta^2) + \theta^2} Z(\theta)^2 dy \right)$$

On obtient finalement :

$$\text{sign}(Z(\theta)) = \begin{bmatrix} -\text{sign}(\theta) & \frac{1 - \text{sign}(\theta)}{1 - \theta} \\ 0 & -1 \end{bmatrix} \quad (3.45)$$

Il est ensuite évident que :

$$\begin{aligned} \text{sign}(Z(\theta)) &= \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \text{ si } \theta > 0 \\ \text{sign}(Z(\theta)) &= \begin{bmatrix} +1 & \frac{2}{1 - \theta} \\ 0 & -1 \end{bmatrix} \text{ si } \theta < 0 \end{aligned}$$

3.2.3.2 Extension au cas multi-paramétrique

En se basant sur la démarche décrite dans la partie précédente, nous définissons une nouvelle représentation similaire à (3.30) pour une matrice à dépendance multi-paramétrique.

$$Z(\theta_*) \triangleq \bar{Z}(\bar{\theta}_* \otimes I_n) \in \mathbb{R}^{n \times n} \quad (3.46)$$

avec :

$$\bar{Z} \in \mathbb{R}^{n \times n w^\gamma}, \quad \bar{\theta}_* = \left[\underbrace{\bar{\theta}_1 \otimes \dots \otimes \bar{\theta}_\gamma}_\gamma \right]^T \in \mathbb{R}^{w^\gamma} \quad \text{et} \quad \bar{\theta}_k = \left[\underbrace{\theta_k^{-l} \dots \theta_k^{-1}}_l \quad \theta_k^0 \quad \underbrace{\theta_k^1 \dots \theta_k^m}_m \right]^T \in \mathbb{R}^w,$$

$$\forall k \in [1, \gamma].$$

Comme pour le cas mono paramétrique, les résultats obtenus dans les Lemmes 3.4 et 3.5 sont repris ici :

$$Z(\theta_*) \cdot V(\theta_*) = \bar{Z}(I_{w^*} \otimes \bar{V})((\bar{\theta}_* \otimes \bar{\theta}_*) \otimes I_n) \quad (3.47)$$

et

$$\text{trace}(Z(\theta_*)) = \begin{bmatrix} \text{trace}(Z_1) & \dots & \text{trace}(Z_{w^*}) \end{bmatrix} \bar{\theta}_* \quad (3.48)$$

Basé sur ces résultats, la fonction signe matricielle d'une dépendance multi-paramétrique peut être décrite selon le corolaire suivant.

Corolaire 3.2 : *Considérons une matrice $Z(\theta)$ donnée par (3.46). Le signe de la matrice $Z(\theta)$ est donné par :*

$$\text{sign}(Z(\theta_*)) = \frac{2}{\pi} Z(\theta_*) \int_0^\infty \left[\sum_{k=1}^n -\frac{p_{k-1}}{p_n} \left(\sum_{i=0}^{n-k} \binom{n-k}{i} y^{2(n-k-i)} Z(\theta_*)^{2i} \right) \right] dy \quad (3.49)$$

$$\text{où : } \begin{cases} p_k = -\frac{1}{k} \sum_{j=1}^k p_{k-j} \left(\sum_{i=0}^j \binom{j}{i} y^{2(j-i)} \text{trace}(Z(\theta_*)^{2i}) \right) \\ p_0 = 1 \end{cases} .$$

3.2.3.3 Stabilité robuste, Equations de Lyapunov et de Riccati

Plusieurs domaines d'applications de la méthode présentée précédemment sont possibles. On notera par exemple le lien avec la stabilité d'une matrice, introduite par (3.48). De plus, la méthode proposée n'est pas seulement utile pour évaluer la stabilité robuste des systèmes linéaires à temps invariant (LTI) mais peut également caractériser l'espace paramétrique exact de stabilité (qui peut être non convexe ou même contenir des espaces disjoints). En fait, $\text{sign}(Z(\theta)) = -I_n$ nous donne un système d'équations non linéaires caractérisant le domaine paramétré de la stabilité robuste.

Une autre application importante des résultats obtenus dans le théorème 3.2 et le corolaire 3.2 est de pouvoir résoudre des équations de Lyapunov de la forme :

$$A(\theta)^T X(\theta) + X(\theta) A(\theta) + Q(\theta) = 0 \quad (3.50)$$

Dans cette partie nous ne présenterons que le cas mono paramétrique dans un souci de clarté. Pour trouver $X(\theta) = X^T(\theta)$ solution de (3.50), considérons la matrice Hamiltonienne correspondante à l'équation de Lyapunov :

$$H(\theta) = \begin{bmatrix} A(\theta) & 0 \\ -Q(\theta) & -A(\theta)^T \end{bmatrix} \quad (3.51)$$

Le signe de cette matrice est défini par :

$$\text{sign}(H(\theta)) = \begin{bmatrix} S_{11}(\theta) & 0 \\ S_{21}(\theta) & S_{22}(\theta) \end{bmatrix} \quad (3.52)$$

En reprenant (3.50) on a :

$$H(\theta) = \begin{bmatrix} I & 0 \\ X(\theta) & I \end{bmatrix} \begin{bmatrix} A(\theta) & 0 \\ 0 & -A(\theta)^T \end{bmatrix} \begin{bmatrix} I & 0 \\ -X(\theta) & I \end{bmatrix} \quad (3.53)$$

Ensuite, en accord avec la propriété (g) sur la fonction signe (chapitre 2) nous obtenons :

$$\text{sign}(H(\theta)) = \begin{bmatrix} I & 0 \\ X(\theta) & I \end{bmatrix} \text{sign} \left(\begin{bmatrix} A(\theta) & 0 \\ 0 & -A(\theta)^T \end{bmatrix} \right) \begin{bmatrix} I & 0 \\ -X(\theta) & I \end{bmatrix}^{-1} \quad (3.54)$$

D'où :

$$\begin{bmatrix} S_{11}(\theta) & 0 \\ S_{21}(\theta) & S_{22}(\theta) \end{bmatrix} = \begin{bmatrix} -I & 0 \\ -2X(\theta) & I \end{bmatrix} \quad (3.55)$$

Finalement, l'unique solution de l'équation de Lyapunov (3.50) est donnée par :

$$X(\theta) = -\frac{1}{2} S_{21}(\theta) \quad (3.56)$$

Exemple 3.7 : Considérons la matrice Hamiltonienne suivante :

$$H(\theta) = \begin{bmatrix} \theta & 0 \\ -1 & -\theta \end{bmatrix}$$

Cette matrice est associée à une simple équation scalaire de Lyapunov. Ecrite sous la forme (3.30), cette matrice s'écrit :

$$H(\theta) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & -1 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \end{bmatrix} \otimes I_2 \right)$$

$$\text{Le carré : } H(\theta)^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \\ \theta^2 \end{bmatrix} \otimes I_2 \right).$$

L'application du théorème 3.2 nous donne :

$$\text{sign}(H(\theta)) = \frac{2}{\pi} H(\theta) \left(\int_0^\infty -\frac{1}{p_2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} y^2 I_2 dy + \int_0^\infty -\frac{1}{p_2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} H(\theta)^2 dy + \int_0^\infty -\frac{p_1}{p_2} I_2 dy \right)$$

avec : $p_1 = -2(y^2 + \theta^2)$, $p_2 = (y^2 + \theta^2)^2$.

On obtiendra finalement : $\text{sign}(H(\theta)) = \frac{2}{\pi} H(\theta) \int_0^\infty \frac{1}{y^2 + \theta^2} I_2 dy$.

Le signe final s'écrit : $\text{sign}(H(\theta)) = \text{sign}(\theta) \begin{bmatrix} 1 & 0 \\ -\frac{1}{\theta} & -1 \end{bmatrix}$.

L'unique solution est donc, reprenant (3.56), $X(\theta) = -\frac{1}{2\theta} > 0$ pour $\theta < 0$.

De manière plus large, il est également possible de résoudre des équations de Riccati paramétrées. Encore une fois, le résultat présenté est dans le cas mono-paramétrique mais facilement extensible au cas multi-paramétrique. L'équation de Riccati considérée dans cette partie est :

$$A^T(\theta)X(\theta) + X(\theta)A(\theta) - X(\theta)B(\theta)R(\theta)^{-1}B(\theta)^T X(\theta) + Q(\theta) = 0 \quad (3.57)$$

Pour trouver $X(\theta) = X^T(\theta)$ solution de l'équation (3.57), nous introduisons la matrice Hamiltonienne suivante :

$$H(\theta) = \begin{bmatrix} A(\theta) & -B(\theta)R(\theta)^{-1}B(\theta)^T \\ -Q(\theta) & -A^T(\theta) \end{bmatrix} \quad (3.58)$$

Nous supposons que la paire $(A(\theta), G(\theta))$ est stabilisable, $G(\theta) = -B(\theta)R(\theta)^{-1}B(\theta)^T$ et la paire $(A(\theta), Q(\theta))$ détectable pour tout θ . Si $H(\theta)$ est supposée être sans valeurs propres à partie réelle nulle pour tout θ , alors l'extension de la méthode présentée par Roberts [Rob80] est directe. On note :

$$\text{sign}(H(\theta)) = \begin{bmatrix} S_{11}(\theta) & S_{12}(\theta) \\ S_{21}(\theta) & S_{22}(\theta) \end{bmatrix} \quad (3.59)$$

La solution $X(\theta)$ de (3.57) est donnée par le système d'équations paramétrées suivant :

$$\begin{pmatrix} S_{12}(\theta) \\ S_{22}(\theta) + I \end{pmatrix} X(\theta) = - \begin{pmatrix} S_{11}(\theta) + I \\ S_{21}(\theta) \end{pmatrix} \quad (3.60)$$

La preuve de (3.60) est équivalente à celle décrite dans [KL95], et rappelée dans le chapitre 2 de ce mémoire, pour le cas constant.

Exemple 3.8 : Considérons la matrice Hamiltonienne suivante :

$$H(\theta) = \begin{bmatrix} \theta & 1 \\ -1 & -\theta \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & -1 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta \end{bmatrix} \otimes I_2 \right)$$

L'application du théorème 3.2 nous donne :

$$\text{sign}(H(\theta)) = \frac{2}{\pi} H(\theta) \int_0^\infty \frac{1}{y^2 + \theta^2 - 1} I_2 dy = \left(\frac{1}{\sqrt{\theta^2 - 1}} \right) \begin{bmatrix} \theta & 1 \\ -1 & -\theta \end{bmatrix}$$

Se basant finalement sur (3.60) la solution de l'équation de Riccati scalaire est :

$$X(\theta) = -\theta - \sqrt{\theta^2 - 1} \text{ avec } \theta < -1.$$

Remarque 3.6 : On notera que la méthode proposée tend à trouver la solution irrationnelle exacte de l'équation de Riccati, chose qui est impossible avec les itérations de Newton (qui convergent vers un approximant rationnel).

Cette partie a fait l'objet d'un papier présenté en conférence [GYC12]. Pour conclure sur cette méthode, le fait d'utiliser des matrices de taille plus grande entraîne des problèmes de calcul d'intégrales. C'est pourquoi nous avons essayé de développer, lors de cette thèse, une généralisation du calcul de ces intégrales. Il nous a néanmoins été possible de généraliser le calcul de la fonction signe matricielle pour des matrices d'ordre 2.

3.3 Cas particulier

3.3.1 Ecriture de la fonction signe d'une matrice d'ordre 2

Le calcul intégral de la fonction signe matricielle n'est pas chose aisée. En effet, cette méthode repose sur le calcul d'intégrales d'éléments rationnels. C'est pourquoi nous avons essayé de trouver un lien entre toutes ces intégrales. Nous avons pu généraliser ceci pour l'ensemble des matrices d'ordre 2 paramétrées. Cette généralisation est faisable pour l'ordre 2, car le type d'intégrales que l'on calcule mène à des termes en arc-tangente, que l'on peut simplifier. En revanche, pour des ordres supérieurs, il apparaîtra des termes logarithmiques, et la généralisation n'a pas encore pu être faite dans ces cas.

Remarque 3.7: Considérons d'abord le cas scalaire, $z \in \mathbb{R}^*$. En reprenant la définition de la forme intégrale de la fonction signe donnée par (3.9) on a :

$$\text{sign}(z) = \frac{2}{\pi} z \int_0^{+\infty} \frac{1}{y^2 + z^2} dy = \frac{2}{\pi} z \left[\frac{\arctan\left(\frac{y}{z}\right)}{z} \right]_0^{+\infty} = \frac{2}{\pi} z \left(\frac{\pi}{2} \frac{\text{sign}(z)}{z} \right) = \text{sign}(z)$$

Considérons maintenant une matrice $Z = \begin{bmatrix} z_1 & z_2 \\ z_3 & z_4 \end{bmatrix}$:

$$\begin{aligned} \text{sign}(Z) &= \frac{2}{\pi} Z \int_0^{+\infty} (y^2 I_2 + Z^2)^{-1} dy \\ &= \frac{2}{\pi} Z \int_0^{+\infty} \frac{1}{y^4 + \alpha y^2 + \beta} \begin{bmatrix} y^2 + z_2 z_3 + z_1^2 & z_2(z_1 - z_4) \\ z_3(z_4 - z_1) & y^2 + z_2 z_3 + z_4^2 \end{bmatrix} dy \end{aligned} \quad (3.61)$$

avec $\alpha = z_1^2 + z_4^2 + 2z_2 z_3$ et $\beta = (z_1^2 + z_2 z_3)(z_4^2 + z_2 z_3) + z_2 z_3 (z_4 - z_1)^2$. On a alors 2 types différents d'intégrales à calculer (on suppose que les constantes sont telles que toutes ces intégrales sont définies) :

$$\begin{aligned} \int_1 = \int_0^{+\infty} \frac{y^2 + \bar{b}}{y^4 + \bar{c}y^2 + \bar{d}} &= \frac{\pi}{2\sqrt{2}\Delta} \left(\frac{2\bar{b} + \Delta - \bar{c}}{\sqrt{\bar{c}} - \Delta} + \frac{-2\bar{b} + \Delta + \bar{c}}{\sqrt{\bar{c}} + \Delta} \right) \\ \int_2 = \int_0^{+\infty} \frac{\bar{b}}{y^4 + \bar{c}y^2 + \bar{d}} &= \frac{\pi\sqrt{2}\bar{b}}{2\Delta} \left(\frac{1}{\sqrt{\bar{c}} - \Delta} - \frac{1}{\sqrt{\bar{c}} + \Delta} \right) \end{aligned}$$

Avec $\Delta = \sqrt{\bar{c}^2 - 4\bar{d}}$, $\bar{c} = \alpha$, $\bar{d} = \beta$ et \bar{b} correspond à chacun des éléments non dépendant de y^2 de la matrice dans l'intégrale.

3.3.2 Résolution d'une équation de Lyapunov d'ordre 2

Cette partie découle de la précédente. Rappelons la forme Hamiltonienne de l'équation de Lyapunov :

$$H_L = \begin{bmatrix} A & 0 \\ -Q & -A^T \end{bmatrix} \quad (3.62)$$

L'idée est de pouvoir exprimer l'inverse de $N = (y^2 I_2 + H^2)$, de la même façon que dans la partie précédente. N s'écrit :

$$N = \begin{bmatrix} y^2 I_2 + A^2 & 0 \\ A^T Q - QA & y^2 I_2 + A^{2T} \end{bmatrix}$$

L'inverse de N s'écrit comme suit :

$$N^{-1} = \begin{bmatrix} (y^2 I_2 + A^2)^{-1} & 0 \\ -\left(y^2 I_2 + A^{2T}\right)^{-1} (A^T Q - QA) (y^2 I_2 + A^2)^{-1} & \left(y^2 I_2 + A^{2T}\right)^{-1} \end{bmatrix}$$

Chaque élément de la matrice N^{-1} peut maintenant être écrit sous la forme (3.61).

Remarque 3.8 : Malheureusement, cette généralisation n'a pas pu être faite pour des matrices de plus grandes dimensions. Il est certes possible de trouver des sous classes de matrices paramétrées pour lesquelles un calcul similaire peut être fait. Néanmoins l'objectif de cette thèse est de proposer des démarches suffisamment génériques, c'est pourquoi, dans un second temps, nous nous sommes intéressés au calcul approché de la fonction signe matricielle, en passant par ce que l'on appelle la séparation de spectre, qui sera présentée dans la prochaine section, et par les méthodes itératives, dans le chapitre 4.

3.4 Fonction signe et spectre matriciel

Cette partie est dédiée à la résolution d'équations de Sylvester paramétrées que l'on retrouve dans le domaine de l'automatique. Les matrices considérées ne sont pas carrées et pas nécessairement de mêmes dimensions (on parle ici des matrices A et B de l'équation *classique* de Sylvester que nous présenterons par la suite) et avec des valeurs propres possiblement communes en fonction des valeurs des paramètres. Une méthode se basant sur la fonction signe est proposée et considère deux cas, la diagonalisation ou bloc-diagonalisation des matrices carrées A et B .

3.4.1 Introduction

L'équation de Sylvester :

$$BX - XA = C \quad (3.63)$$

avec $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times m}$ et $X \in \mathbb{R}^{m \times n}$, a été abondamment étudiée dans la littérature. Des algorithmes standards de résolution de cette équation sont basés sur la méthode de Bartels-Stewart [BS72] et la méthode de Hessenberg-Schur [GNV-L79]. On notera également la décomposition spectrale [JL82] ou encore la solution polynomiale donnée dans [HC06]. On a d'un autre côté la méthode se basant sur la fonction signe matricielle [Rob80] dans le cadre d'équations de Sylvester dites « stables » (i.e. équivalentes à des équations de Lyapunov). Récemment, une fonction signe modifiée a été proposée dans [LBW10] pour résoudre des équations de Sylvester généralisées apparaissant dans des problèmes d'analyse et de contrôle de systèmes descripteurs.

Dans cette partie, une méthode de fonction signe matricielle est proposée afin de résoudre une équation de Sylvester paramétrique de la forme :

$$B_\theta(\theta) X(\theta) - X(\theta) A_\theta(\theta) = C_\theta(\theta), \quad \theta \in \Omega_\theta \subset \mathbb{R}^r \quad (3.64)$$

L'équation de Sylvester (3.63) a beaucoup d'applications dans divers domaines, par exemple, en commande (commande et filtrage, réduction de modèles ; voir [ZDG96]) et en traitement du signal. Plus particulièrement, en commande, des cas particuliers de grand intérêt concernent des problèmes de régulation asymptotiques [KS09] et étendus ([SSS00], [FYC11] et références incluses) qui mènent à la résolution d'équations de Sylvester de la forme (3.63). Il est bien établi dans la littérature quand et comment ce problème peut être résolu sous forme constante (c.-à-d. avec les matrices A et B à coefficients constants). Le cas paramétrique, en revanche, est rarement mentionné, sauf dans [KS11] et peut donc être une application importante pour la méthode basée sur la fonction signe matricielle présentée dans cette partie.

L'idée sous-jacente de cette méthode se base sur deux résultats, une méthode de fonction signe matricielle pour le calcul de valeurs propres d'une matrice [BD73] et la solution d'une équation de Sylvester à l'aide d'une décomposition spectrale de matrices [JL82]. Plusieurs exemples pour extraire les valeurs propres d'une matrices utilisant la fonction signe matricielle existent dans la littérature. Voir par exemple [Sti91] qui donne un éclairage différent sur l'utilisation de la fonction signe matricielle dans ce contexte. L'utilisation du résultat [BD73] a pour but d'étendre la méthode au cas paramétrique (3.64).

3.4.2 Un algorithme pour la séparation de spectre

Dans cette partie, une nouvelle méthode est proposée pour résoudre une équation de Sylvester de la forme (3.63) à l'aide de la fonction signe matricielle. Cette méthode consiste en trois étapes. La fonction signe matricielle est utilisée afin de déterminer les valeurs propres des matrices A et B . Ensuite une décomposition spectrale de ces matrices (calcul de projecteurs) est réalisée. Finalement, à l'aide des 2 étapes précédentes, nous présentons la solution explicite X de l'équation de Sylvester.

L'idée sous-jacente de cette méthode est basée sur deux résultats, comme indiqué dans l'introduction, le calcul de la fonction signe matricielle pour les valeurs propres d'une matrice [BD73] et la solution d'une équation de Sylvester basée sur la décomposition spectrale de matrices [JL82]. Le but principal, comme dans l'ensemble de ces travaux de thèse, est de proposer une méthode pouvant être étendue au cas paramétré (3.64).

3.4.2.1 Le cas de matrices diagonalisables à coefficients constants

La démarche sera présentée à chaque fois sous forme de trois étapes distinctes.

Etape 1 : Séparation du spectre

Considérons $Z \in \mathbb{R}^{n \times n}$, une matrice carrée ayant la forme canonique de Jordan suivante :

$$Z = ML_Z M^{-1} \quad (3.65)$$

avec L_Z une matrice triangulaire composée de blocs de Jordan associés aux valeurs propres de Z , et M la matrice des vecteurs propres de Z . La matrice signe de Z est alors donnée par :

$$\text{sign}(Z) = M \text{sign}(L_Z) M^{-1} \quad (3.66)$$

Dans cette sous-section, la matrice Z est supposée être diagonalisable avec $\sigma^+(Z) \neq \emptyset$ et $\sigma^-(Z) \neq \emptyset$, ce qui signifie qu'il doit y avoir au moins une valeur propre positive et une négative. $\sigma(Z) = \{\lambda_{Z_0}, \dots, \lambda_{Z_{n-1}}\}$ définit le spectre de Z .

Reprenant [BD73], une première séparation de spectre peut être obtenue avec V_0 donné par :

$$V_0 = \text{sign}(Z) + \text{sign}(L_Z), \quad V_0^{-1} Z V_0 = \begin{bmatrix} Z_1 & 0 \\ 0 & Z_2 \end{bmatrix} \quad (3.67)$$

Les valeurs propres de Z_1 et Z_2 sont à parties réelles positives et négatives respectivement. Cette séparation de spectre peut être complétée en calculant les matrices \bar{Z}_1 et \bar{Z}_2 comme suit :

$$\bar{Z}_k = Z_k - \left(\frac{\text{trace}(Z_k)}{\dim(Z_k)} \right) I_{\dim(Z_k)}, \quad k = 1, 2 \quad (3.68)$$

Ensuite, observant le signe des matrices obtenues \bar{Z}_1 et \bar{Z}_2 , la partition de Z_1 et Z_2 en matrices bloc-diagonales est obtenue grâce à :

$$V_k = \text{sign}(\bar{Z}_k) + \text{sign}(L_{\bar{Z}_k}), \quad k = 1, 2 \quad (3.69)$$

Les itérations peuvent continuer ainsi jusqu'à ce que toutes les sous-matrices soient réduites à une forme de Jordan.

Remarque 3.9 : Les matrices déficientes requièrent une étape supplémentaire. En effet, si $\text{sign}(Z) = I_n$, cela montre que les valeurs propres de Z sont égales ou de même signe. Pour déterminer la répétition de ces valeurs propres, il est nécessaire de calculer :

$$\bar{Z} = Z - \left(\frac{\text{trace}(Z)}{n} \right) I_n \quad (3.70)$$

Si \bar{Z} est singulière, cela veut dire que les valeurs propres sont similaires et donc, une nouvelle étape est nécessaire et sera décrite dans la sous-section suivante.

Etape 2 : Calcul des projecteurs

Les matrices E_j de la décomposition spectrale $Z = \sum_{j=0}^{n-1} \lambda_{Z_j} E_j$, où les λ_{Z_j} sont les valeurs propres distinctes non-nulles de Z sont telles que :

$$E_j^2 = E_j, \sum_{j=0}^{n-1} E_j = I, E_j E_k = 0, j \neq k \quad (3.71)$$

Une manière de calculer ces projecteurs est la suivante (voir [BD73] par exemple) :

$$E_j = \left(\prod_{i=0}^{n-1} \left(1 - \frac{\lambda_{Z_j}}{\lambda_{Z_i}} \right) \right)^{-1} \cdot \prod_{i=0}^{n-1} \left(I_n - \frac{Z}{\lambda_{Z_i}} \right), i \neq j \quad (3.72)$$

Remarque 3.10 : Notons que l'équation (3.72) utilise des produits de matrices et peut donc être calculée au cas paramétrique, en exploitant la classe de matrices paramétrées déjà introduite dans les précédents chapitres et rappelée dans la prochaine section.

Etape 3 : La solution analytique de l'équation de Sylvester (3.63)

Posons $A = \sum_{j=0}^{n-1} \lambda_{A_j} E_j$ et $B = \sum_{k=0}^{m-1} \lambda_{B_k} F_k$ les décompositions spectrales de A et B , où λ_{A_j} et λ_{B_k} sont respectivement leurs valeurs propres distinctes non-nulles. Les matrices E_j et F_k sont idempotentes et vérifient les propriétés données dans (3.71).

Théorème 3.3 : [JL82] *L'équation de Sylvester (3.63) avec des matrices non déficientes à coefficients constants admet une solution si et seulement si pour tout i et k tels que $\lambda_{A_j} = \lambda_{B_k}$ alors $F_k C E_j = 0$. La solution est alors donnée par :*

$$X = \sum_{k=0}^{m-1} \sum_{j=0}^{n-1} \frac{1}{(\lambda_{B_k} - \lambda_{A_j})} F_k C E_j \quad (3.73)$$

Exemple 3.9 :

Considérons l'équation (3.63) avec les données suivantes :

$$A = \begin{bmatrix} 2.5 & -1 \\ 0 & -0.5 \end{bmatrix}, B = T \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -3 \end{bmatrix} T^{-1}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{bmatrix}, T = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 2 & 4 \\ 1 & 5 & 3 \end{bmatrix}$$

Reprenons les étapes décrites précédemment.

Etape 1 : Séparation de spectre

Calculons d'abord $V_{0,A}$ pour la matrice A :

$$V_{0,A} = \text{sign}(A) + \text{sign}(L_A)$$

$\text{sign}(A)$ est calculé en utilisant les itérations de Newton. Ensuite, $\text{sign}(L_A)$ est en fait le signe de chaque valeur propre, et regardant la trace de $\text{sign}(A)$:

$$V_{0,A} = \begin{bmatrix} 1 & -0.6667 \\ 0 & -1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 2 & -0.6667 \\ 0 & -2 \end{bmatrix}$$

Il vient ensuite $V_{0,A}^{-1}AV_{0,A} = \begin{bmatrix} 2.5 & 0 \\ 0 & -0.5 \end{bmatrix}$ et $\sigma(A) = \{2.5, -0.5\}$.

Calculons maintenant $V_{0,B}$ pour la matrice B :

$$V_{0,B} = \text{sign}(B) + \text{sign}(L_B) = \begin{bmatrix} -6 & 3 & 2 \\ -10.6667 & 6 & 2.6667 \\ -8 & 3 & 2 \end{bmatrix}$$

Ensuite, $V_{0,B}^{-1}BV_{0,B} = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 \\ 1.5556 & 1 & 0 \\ 0 & 0 & -3 \end{bmatrix},$

et à l'aide de (3.68) : $\bar{B}_1 = \begin{bmatrix} 0.5 & 0 \\ 1.5556 & -0.5 \end{bmatrix}.$

$$V_{1,B} = \text{sign}(\bar{B}_1) + \text{sign}(L_{\bar{B}_1}) = \begin{bmatrix} 2 & 0 \\ 3.1112 & -2 \end{bmatrix}$$

Ce qui nous donne finalement : $V_{1,B}^{-1}B_1V_{1,B} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}.$ Le spectre est donné par $\sigma(B) = \{2, 1, -3\}.$

Etape 2 : Calcul des projecteurs

Grâce à (3.72), E_0 , E_1 , F_0 , F_1 et F_2 sont donnés par :

$$E_0 = \begin{bmatrix} 1 & -0.333 \\ 0 & 0 \end{bmatrix}, E_1 = \begin{bmatrix} 0 & 0.333 \\ 0 & 1 \end{bmatrix}, F_0 = \begin{bmatrix} -0.667 & 0 & 0.667 \\ -0.667 & 0 & 0.667 \\ -1.667 & 0 & 1.667 \end{bmatrix}, F_1 = \begin{bmatrix} -2.333 & 1.5 & 0.333 \\ -4.667 & 3 & 0.667 \\ -2.333 & 1.5 & 0.333 \end{bmatrix},$$

$$F_2 = \begin{bmatrix} 4 & -1.5 & -1 \\ 5.333 & -2 & -1.333 \\ 4 & -1.5 & -1 \end{bmatrix}$$

Etape 3 : La solution analytique de l'équation de Sylvester (3.63)

Les matrices A et B n'ont pas de valeurs propres communes et l'équation (3.73) amène à l'unique solution de (3.63) :

$$X = \begin{bmatrix} 2.1616 & 3.4747 & 4.1616 \\ 1.5165 & 2.2936 & 1.5165 \end{bmatrix}^T$$

3.4.2.1 Le cas de matrices déficientes à coefficients constants

Etape 1 : Séparation du spectre

Une matrice déficiente $Z \in \mathbb{R}^{n \times n}$ requiert une modification de la procédure proposée, comme noté en Remarque 4.1. Dans cette sous-section, et sans perte de généralité, Z est supposée n'avoir que des valeurs propres répétées notées λ_z . Si ce n'est pas le cas, une décomposition spectrale peut être effectuée comme décrit précédemment.

Comme dans ce cas $\text{sign}(Z) = \pm I_n$, il est facile de voir :

$$\lambda_z = \frac{\text{trace}(Z)}{n} \quad (3.74)$$

Notons que dans ce cas, \bar{Z} , calculée en utilisant l'équation (3.70), est singulière et n'admet donc pas de signe matriciel.

Etape 2 : Calcul des projecteurs

Dans cette étape, la matrice $Z \in \mathbb{R}^{n \times n}$ est supposée être déficiente mais pas complètement.

Les matrices E_j de la décomposition spectrale $Z = \sum_{j=0}^{n_1-1} \lambda_{z_j} E_j + N_j$, où λ_{z_j} sont les valeurs propres non-nulles de la matrice Z , sont idempotentes et vérifient les propriétés (3.71). Les matrices N_j sont nilpotentes et vérifient, elles, les conditions suivantes :

$$E_j N_j = N_j E_j = N_j, E_j N_k = N_j N_k = 0, \forall j \neq k \quad (3.75)$$

Ces matrices peuvent être calculées de la manière suivante :

$$N_j = (A - \lambda_{A_j} I_n) E_j \quad (3.76)$$

Notons que le calcul des λ_{Z_j} est possible en se basant sur la procédure présentée précédemment. En outre, le calcul des matrices E_j prend en compte la fonction :

$$f(Z) = \sum_{j=0}^{n-1} \sum_{p=0}^{p_j-1} \frac{f^{(p)}}{p!} (Z - \lambda_{Z_j} I_n)^p E_j \quad (3.77)$$

où $p_j = \text{index}(\lambda_{Z_j})$, $j = 0, 1, \dots, (n-1)$. Les projecteurs spectraux E_j peuvent être calculés itérativement grâce à un choix judicieux de f donné par :

$$f(\lambda) = \left\{ 1, (\lambda - \lambda_{A_1})^{p_1}, (\lambda - \lambda_{A_1})^{p_1} (\lambda - \lambda_{A_2})^{p_2}, \dots, (\lambda - \lambda_{A_1})^{p_1} \dots (\lambda - \lambda_{A_{n_1}})^{p_{n_1}} \right\} \quad (3.78)$$

Etape 3 : La solution analytique de l'équation de Sylvester (3.63)

Posons $A = \sum_{j=0}^{n_1-1} \lambda_{A_j} E_j + N_j$ et $B = \sum_{k=0}^{m_1-1} \lambda_{B_k} F_k + M_k$ les décompositions spectrales de A et B , λ_{A_j} et λ_{B_k} les valeurs propres non-nulles de A et B respectivement. Notons également $n_1 \leq n$ et $m_1 \leq m$ le nombre de valeurs propres distinctes de A et B respectivement.

Théorème 3.4 : [JL82] L'équation de Sylvester (3.63) avec des matrices déficientes à coefficients constants admet :

$$X = \sum_{k=0}^{m_1-1} \sum_{j=0}^{n_1-1} \frac{1}{(\lambda_{B_k} - \lambda_{A_j})} F_k C E_j \quad (3.79)$$

comme solution si il existe pour tout i et k tels que $\lambda_{A_j} = \lambda_{B_k}$ et $F_k C N_j = M_k C E_j$,

$$F_k C E_j = 0 \text{ et } \sum_{k=0}^{m_1-1} \sum_{j=0}^{n_1-1} \frac{1}{(\lambda_{B_k} - \lambda_{A_j})} (F_k C N_j - M_k C E_j) = 0.$$

Exemple 3.10 : Considérons l'équation de Sylvester (3.63) avec les données suivantes :

$$A = \begin{bmatrix} -2 & -1 \\ 0 & -1 \end{bmatrix}, B = \begin{bmatrix} 2 & -1 & 1 \\ 3 & 3 & -2 \\ 4 & 1 & 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Reprenons les nouvelles étapes décrites dans cette partie.

Étape 1 : Séparation du spectre

$\text{sign}(A) = -I_2$ et $\text{sign}\left(A - \left(\frac{\text{trace}(A)}{2}\right)I_2\right) \neq -I_2$, donc la matrice A a deux valeurs propres distinctes à partie réelle négative. Le calcul de \bar{A} et V_0 donne :

$$\bar{A} = A - \left(\frac{\text{trace}(A)}{2}\right)I_2, \quad V_0 = \text{sign}(\bar{A}) + \text{sign}(L_{\bar{A}}) = \begin{bmatrix} -2 & -2 \\ 0 & 2 \end{bmatrix}$$

$$\text{D'où } V_0^{-1}AV_0 = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix}.$$

Pour la matrice B , nous obtenons :

$$\text{sign}(B) = I_3 \text{ et } \bar{B} = B - \left(\frac{\text{trace}(B)}{3}\right)I_3, \quad \text{sign}(\bar{B}) = \begin{bmatrix} 0 & -1 & 1 \\ 0.5 & -1.5 & 0.5 \\ 1.5 & -1.5 & 0.5 \end{bmatrix} \neq I_3.$$

Dans ce cas, le calcul de V_0 se fait par :

$$V_0 = \text{sign}(\bar{B}) + \text{sign}(L_{\bar{B}}) = \begin{bmatrix} 1 & -1 & 1 \\ 0.5 & -2.5 & 0.5 \\ 1.5 & -1.5 & -0.5 \end{bmatrix}$$

$$\text{Et amène à : } V_0^{-1}BV_0 = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3.5 & -2.5 \\ 0 & 2.5 & -1.5 \end{bmatrix}.$$

Nous pouvons directement voir ici que 3 est une valeur propre de B . La matrice

$$B_2 = \begin{bmatrix} 3.5 & -2.5 \\ 2.5 & -1.5 \end{bmatrix} \text{ est telle que } \text{sign}(B_2) = I_2 \text{ et } \bar{B}_2 = B_2 - \left(\frac{\text{trace}(B_2)}{2}\right)I_2 = \begin{bmatrix} 2.5 & -2.5 \\ 2.5 & -2.5 \end{bmatrix} \text{ est}$$

singulière. Ensuite, B_2 a la valeur répétée : $\lambda_{B_2} = \frac{\text{trace}(B_2)}{2} = 1$.

Étape 2 : Calcul des projecteurs

Comme λ_{A_j} et λ_{B_k} sont connues, nous pouvons calculer les projecteurs E_j et F_k . La matrice A est diagonalisable donc le calcul de E_j peut être effectué à l'aide de (3.72) et donne :

$$E_0 = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \quad E_1 = \begin{bmatrix} 0 & -1 \\ 0 & 1 \end{bmatrix}$$

Pour calculer les F_k considérons la fonction suivante :

$$f(B) = f(3)F_0 + f(1)F_1 + f'(1)(B - I_m)F_1$$

avec les choix suivants :

$$\begin{cases} f(z) = 1 & \Rightarrow F_1 = F_0 - I_3 \\ f(z) = (z-1)^2 & \Rightarrow F_0 = \frac{1}{4}(B - I_3)^2 \end{cases}$$

Les projecteurs spectraux sont finalement obtenus comme suit :

$$F_0 = \frac{1}{4} \begin{bmatrix} 2 & -2 & 2 \\ 1 & -1 & 1 \\ 3 & -3 & 3 \end{bmatrix}, \quad F_1 = \frac{1}{4} \begin{bmatrix} 2 & 2 & -2 \\ -1 & 5 & -1 \\ -3 & 3 & 1 \end{bmatrix}$$

Étape 3 : La solution analytique de l'équation de Sylvester (3.63)

Se basant sur le théorème 3.4, l'équation (3.79) nous donne la solution suivante :

$$X = \begin{bmatrix} 0 & 0.333 & 0.333 \\ 0 & 0.333 & 0.333 \end{bmatrix}^T$$

3.4.3 Une fonction signe matricielle pour la solution d'une équation de Sylvester paramétrique

Dans cette section, une extension au cas paramétrique (3.64) de la procédure précédemment proposée est présentée. Dans ce but, nous introduisons la classe suivante de matrices :

$$Z(\theta) \triangleq Z_e(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}, \quad \theta \in \Omega_\theta \quad (3.80)$$

avec $\bar{\theta} = [\bar{\theta}_1 \otimes \dots \otimes \bar{\theta}_\gamma]^T \in \mathbb{R}^{w^\gamma}$ et $\bar{\theta}_k = [\theta_k^{-l} \quad \dots \quad \theta_k^{-l} \quad 1 \quad \theta_k^1 \quad \dots \quad \theta_k^m]^T \in \mathbb{R}^w$,
 $\forall k \in [1, \gamma]$.

Dans la suite, $Z(\theta)$ est supposée non singulière. De nouvelles opérations sont introduites ici et basées sur la définition (3.80) et décrites dans le lemme suivant.

Lemme 3.4 : Posons $Z(\theta) = Z_e(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$, $V(\theta) = V_e(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$, une partition de Z_e donnée par $Z_e = [Z_1 \ \cdots \ Z_w]$, $Z_{i \in [1,w]} \in \mathbb{R}^{n \times n}$ et $\bar{\theta} = [\bar{\theta}_1 \otimes \cdots \otimes \bar{\theta}_\gamma]^T$. Alors, la somme $Z(\theta) + V(\theta)$, le produit $Z(\theta)V(\theta)$ et la trace de $Z(\theta)$ donnent :

$$Z(\theta) + V(\theta) = (Z_e + V_e)(\bar{\theta} \otimes I_n) \quad (3.81)$$

$$Z(\theta) \cdot V(\theta) = Z_e(I_{m'} \otimes V_e)((\bar{\theta} \otimes \bar{\theta}) \otimes I_n) \quad (3.82)$$

$$\text{trace}(Z(\theta)) = [\text{trace}(Z_1) \ \cdots \ \text{trace}(Z_w)] \bar{\theta} \quad (3.83)$$

Dans cette partie, plutôt que d'utiliser les itérations de Newton, qui donnent une approximation rationnelle de $\text{sign}(Z(\theta))$, nous préférons nous baser sur la définition intégrale pour le calcul de la fonction $\text{sign}(Z(\theta))$ qui calcule la valeur exacte du signe.

Hypothèse (H1) : la matrice $Z(\theta)$, $\theta \in \Omega_\theta$ est soit diagonalisable soit bloc-diagonalisable $\forall \theta \in \Omega_\theta$.

(H1) signifie que $Z(\theta)$ garde la même classe de décomposition spectrale, $\forall \theta \in \Omega_\theta$. En outre, sous l'hypothèse (H1), le calcul de la décomposition spectrale de $Z(\theta)$ suivra les mêmes routines que celles décrites précédemment.

Considérons maintenant l'équation de Sylvester (3.64) avec les matrices $A_\theta(\theta)$, $B_\theta(\theta)$ et $C_\theta(\theta)$ de la forme (3.80).

Théorème 3.5 : Sous l'hypothèse (H1), l'équation de Sylvester paramétrique (3.64) admet la solution donnée par :

$$X(\theta) = \sum_{k=0}^{m-1} \sum_{j=0}^{n-1} \frac{1}{(\lambda_{B_{\theta_k}}(\theta) - \lambda_{A_{\theta_j}}(\theta))} F_k(\theta) C_\theta(\theta) E_j(\theta) \quad (3.84)$$

si et seulement si pour tout j, k et $\theta_0 \in \Omega_\theta$ tels que $\lambda_{A_{\theta_j}}(\theta_0) = \lambda_{B_{\theta_k}}(\theta_0)$, l'équation suivante est vérifiée $F_k(\theta_0) C_\theta(\theta_0) E_j(\theta_0) = 0$ avec $E_j(\theta)$, $F_k(\theta)$ vérifiant les conditions (3.71) et $\lambda_{A_{\theta_j}}, \lambda_{B_{\theta_k}}$ sont les valeurs propres non-nulles des matrices A_θ et B_θ .

Preuve : (Nécessité) Supposons que $X(\theta)$ est solution de l'équation (3.64) et qu'il existe j, k et $\theta_0 \in \Omega_\theta$ tels que $\lambda_{A_{\theta_j}}(\theta_0) = \lambda_{B_{\theta_k}}(\theta_0)$ alors :

$$\sum_{k=0}^{m-1} \lambda_{B_{\theta_k}}(\theta_0) F_k(\theta_0) X(\theta_0) - \sum_{j=0}^{n-1} \lambda_{A_{\theta_j}}(\theta_0) X(\theta_0) E_j(\theta_0) = C_\theta(\theta_0)$$

En utilisant les propriétés d'orthogonalité des matrices F_k et E_j nous avons :

$$\lambda_{B_{\theta_k}}(\theta_0) F_k(\theta_0) X(\theta_0) E_j(\theta_0) - \lambda_{A_{\theta_j}}(\theta_0) F_k(\theta_0) X(\theta_0) E_j(\theta_0) = F_k(\theta_0) C_\theta(\theta_0) E_j(\theta_0)$$

Ensuite, $\lambda_{A_{\theta_j}}(\theta_0) = \lambda_{B_{\theta_k}}(\theta_0)$ implique que $F_k(\theta_0) C_\theta(\theta_0) E_j(\theta_0) = 0$. La partie suffisante est faite par de simples opérations. ■

Exemple 3.11 : Considérons l'équation matricielle (3.64) avec les données suivantes :

$$A_\theta = 1, B_\theta(\theta) = \begin{bmatrix} \theta & 1 \\ -1 & -\theta \end{bmatrix}, C_\theta = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \Omega_\theta =]1, +\infty[$$

Etape 1 : Séparation du spectre

Le calcul de $\text{sign}(B_\theta(\theta))$ en utilisant le Théorème 4.3 mène à :

$$\text{sign}(B_\theta(\theta)) = \frac{2}{\pi} B_\theta(\theta) \int_0^\infty \frac{1}{y^2 + \theta^2 - 1} I_2 dy$$

$$\text{Et finalement, } \text{sign}(B_\theta(\theta)) = \left(\frac{1}{\sqrt{\theta^2 - 1}} \right) \begin{bmatrix} \theta & 1 \\ -1 & -\theta \end{bmatrix}.$$

Puis, comme pour le premier exemple, $V_0(\theta)$ est donné par :

$$V_0(\theta) = \text{sign}(B_\theta(\theta)) + \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$\text{Et } V_0(\theta)^{-1} B_\theta(\theta) V_0(\theta) = \sqrt{\theta^2 - 1} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Maintenant, les valeurs propres de $B_\theta(\theta)$ sont directement $\lambda_{B_{\theta_{1,2}}} = \pm \sqrt{\theta^2 - 1}$ et $\lambda_{B_{\theta_1}} \neq \lambda_{B_{\theta_2}}$ pour tout $\theta \in \Omega_\theta$.

Etape 2 : Calcul des projecteurs

Le calcul de E_j et F_k en utilisant l'équation (3.72) mène à :

$$E_0 = 1, F_0(\theta) = \frac{1}{2} \left(I_2 + \frac{1}{\sqrt{\theta^2 - 1}} B_\theta(\theta) \right) \text{ et } F_1(\theta) = \frac{1}{2} \left(I_2 - \frac{1}{\sqrt{\theta^2 - 1}} B_\theta(\theta) \right)$$

Étape 3 : La solution analytique de l'équation de Sylvester (3.64)

Pour $\theta = \sqrt{2}$, nous avons $\lambda_{B_{\theta_0}} = \lambda_{A_{\theta_0}} = 1$ et $F_0(\sqrt{2})C_\theta E_0 = \begin{bmatrix} 1.207 \\ -0.5 \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. Finalement, se basant sur le Théorème 3.5, l'équation de Sylvester paramétrique admet la solution suivante :

$$X_\theta(\theta) = \frac{1}{\sqrt{\theta^2 - 1} - 1} F_0(\theta) C_\theta E_0 - \frac{1}{\sqrt{\theta^2 - 1} + 1} F_1(\theta) C_\theta E_0$$

$$X_\theta(\theta) = \begin{bmatrix} \frac{1}{\theta^2 - 2} \\ \frac{1 - \theta}{\theta^2 - 2} \end{bmatrix}, \forall \theta \in \Omega_\theta / \theta \neq \sqrt{2}.$$

Le résultat suivant est l'extension du Théorème 3.4.

Théorème 3.6 : Sous l'hypothèse (H1), l'équation de Sylvester paramétrique (3.64) admet la solution donnée par :

$$X(\theta) = \sum_{k=0}^{m_1-1} \sum_{j=0}^{n_1-1} \frac{1}{\left(\lambda_{B_{\theta_k}}(\theta) - \lambda_{A_{\theta_j}}(\theta) \right)} F_k(\theta) C_\theta(\theta) E_j(\theta) \quad (3.85)$$

si pour tout j, k et $\theta_0 \in \Omega_\theta$ tels que $\lambda_{A_{\theta_j}}(\theta_0) = \lambda_{B_{\theta_k}}(\theta_0)$ les conditions suivantes sont vérifiées :

$$\begin{aligned} F_k(\theta_0) C_\theta(\theta_0) N_j(\theta_0) &= M_k(\theta_0) C_\theta(\theta_0) E_j(\theta_0) \\ F_k(\theta_0) C_\theta(\theta_0) E_j(\theta_0) &= 0 \end{aligned} \quad (3.86)$$

et

$$\sum_{k=0}^{m_1-1} \sum_{j=0}^{n_1-1} \frac{1}{\left(\lambda_{B_{\theta_k}}(\theta) - \lambda_{A_{\theta_j}}(\theta) \right)} \cdot \left(F_k(\theta) C_\theta(\theta) N_j(\theta) - M_k(\theta) C_\theta(\theta) E_j(\theta) \right) = 0 \quad (3.87)$$

avec $E_j(\theta)$, $F_k(\theta)$ vérifiant les conditions (3.71), $N_j(\theta)$, $M_k(\theta)$ vérifiant les conditions (3.75) et $\lambda_{A_{\theta_j}}$, $\lambda_{B_{\theta_k}}$ les valeurs propres non-nulles de A_θ et B_θ .

Idée de la preuve : Montrons que lorsque l'on a une valeur propre commune pour la condition (3.86), alors $X(\theta)$ donné par (3.85) est une solution de l'équation (3.64).

De simples calculs nous mènent à :

$$A_\theta X - XB_\theta = \sum_{k=0}^{m_1-1} \sum_{j=0}^{n_1-1} \frac{1}{(\lambda_{B_{\theta_k}} - \lambda_{A_{\theta_j}})} F_k C_\theta E_j + \sum_{k=0}^{m_1-1} \sum_{j=0}^{n_1-1} \frac{1}{(\lambda_{B_{\theta_k}} - \lambda_{A_{\theta_j}})} (F_k C_\theta N_j - M_k C_\theta E_j)$$

$$A_\theta X - XB_\theta = \sum_{k=0}^{m_1-1} F_k C_\theta \sum_{j=0}^{n_1-1} E_j = C_\theta. \quad \blacksquare$$

Exemple 3.12 : Considérons l'équation matricielle (3.64) avec les données suivantes :

$$A_\theta(\theta) = \begin{bmatrix} \frac{1}{\theta} - 2 & 1 \\ -4 & \frac{1}{\theta} + 2 \end{bmatrix}, \quad B_\theta = 2, \quad C_\theta = \begin{bmatrix} 2(\theta-1) \\ 1-\theta \end{bmatrix}^T, \quad \Omega_\theta =]0, +\infty[$$

Etape 1 : Séparation du spectre

Le calcul de $\text{sign}(A_\theta(\theta))$ en utilisant mène à :

$$\text{sign}(A_\theta(\theta)) = \frac{2}{\pi} A_\theta(\theta) \cdot \int_0^\infty \frac{\theta^2}{(\theta^2 y^2 + 1)^2} \begin{bmatrix} \theta^2 y^2 + 4\theta + 1 & -2\theta \\ 8\theta & \theta^2 y^2 - 4\theta + 1 \end{bmatrix} dy$$

D'où $\text{sign}(A_\theta(\theta)) = \text{sign}(\theta) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ comme $\theta > 0$ et

$\bar{A}_\theta(\theta) = A_\theta(\theta) - \frac{\text{trace}(A_\theta(\theta))}{2} = \begin{bmatrix} -2 & 1 \\ -4 & 2 \end{bmatrix}$ est une matrice singulière. Les valeurs propres

de $A_\theta(\theta)$ sont directement obtenues :

$$\lambda_{A_{\theta_1,2}}(\theta) = \frac{\text{trace}(A_\theta(\theta))}{2} = \frac{1}{\theta}, \quad \theta \in \Omega_\theta.$$

Etape 2 : Calcul des projecteurs

Le calcul de E_j et F_k en utilisant (3.76), (3.77) et (3.78) donnent : $F_0 = 1$, $E_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ et

$$N_0 = \begin{bmatrix} -2 & 1 \\ -4 & 2 \end{bmatrix}.$$

Etape 3 : La solution analytique de l'équation de Sylvester (3.64)

Pour $\theta=1$, nous avons $\lambda_{B_{\theta_0}} = \lambda_{A_{\theta_0}} = 1$ et $F_0 C_{\theta}(1) N_0 = M_0 C_{\theta}(1) E_0$, $F_0 C_{\theta}(1) E_0 = 0$ vérifiant la condition suffisante (3.87).

Finalement,

$$X_{\theta}(\theta) = \left(\frac{1}{2-1/\theta} \right) F_0 C_{\theta} E_0 = \begin{bmatrix} \frac{2\theta}{\theta^2 + \theta + 1} & \frac{-\theta^2 + \theta}{\theta^3 - 1} \end{bmatrix}$$

est une solution possible pour $\theta \in \Omega_{\theta} / \theta \neq 1$.

3.5 Conclusion

Dans un premier temps, ce chapitre s'est focalisé sur le calcul de la fonction signe matricielle via la définition intégrale. Une méthode générique a été recherchée et développée dans le cas d'ordre 2. Une généralisation semble possible à travers un grand nombre d'exemples traités mais n'a pas été formalisée. Néanmoins, cette démarche originale peut s'avérer nécessaire pour certaines applications où la solution analytique est souhaitable. Ce chapitre s'est également concentré sur la résolution d'équations de Sylvester paramétrées, où les matrices A et B de l'équation classique de Sylvester ne sont pas de même dimension et pouvant avoir des valeurs propres communes en fonction des valeurs des paramètres. Une méthode a été proposée considérant deux cas, la diagonalisation ou bloc-diagonalisation des matrices carrées A et B . L'idée sous-jacente de cette méthode s'est basée sur deux résultats, une méthode de fonction signe matricielle pour le calcul de valeurs propres d'une matrice et la solution d'une équation de Sylvester à l'aide d'une décomposition spectrale de matrices. Cette partie a fait l'objet d'une publication en conférence [GYC14a]. Néanmoins, trouver le signe exact, c'est-à-dire, se basant sur l'écriture intégrale de la fonction signe matricielle ne nous permet pas d'étudier le cas de matrices de grandes dimensions. C'est pourquoi le chapitre 4 est dédié aux calculs de signe de manière itérative.

Chapitre 4 :

Méthodes itératives

Rien n'est plus proche du vrai que le faux.

Albert Einstein

Ce chapitre a pour but d'aborder la fonction signe matricielle de manière itérative pour la résolution d'équations de Lyapunov et de Riccati paramétrées. Trois démarches sont proposées. Dans un premier temps nous proposons des approximations rationnelles inspirées des polynômes de Laurent pour le calcul de la fonction signe matricielle introduits dans le Chapitre 2. Dans une seconde partie, une combinaison entre la fonction signe matricielle et les inversions de matrices paramétrées basées sur les DFT est présentée. Finalement, le cas de la fonction signe matricielle étendu sera présenté afin de faire l'étude du domaine de stabilité.

4.1 Approximation de la fonction signe matricielle par les polynômes de Laurent

Dans cette partie, nous considérons le problème de résolution d'équations de Riccati paramétrées. Nous présenterons un algorithme itératif utilisant principalement des multiplications et additions de matrices permettant d'approximer la solution de ces équations. Le résultat est appliqué d'abord au cas constant avant d'être étendu au cas paramétré. La méthode se base sur deux résultats : une caractérisation palindromique quadratique polynomiale des fonctions signe matricielle et racine matricielle, et une représentation spécifique de matrices paramétrées.

4.1.1 Introduction

Les équations algébriques de Riccati paramétrées (ARE) sont souvent utilisées dans la synthèse de régulateur comme, par exemple, la commande à séquençement de gain [AG95], [SR99], la synthèse de loi de commande devant répondre à des objectifs contradictoires (critères de performance et de robustesse par exemple) [DSF+05], [DSF+03]. Le but principal de cette partie est de proposer un algorithme de complexité réduite permettant de déterminer des solutions approchées à de tels problèmes, avec une précision arbitraire. Il est basé sur la fonction signe matricielle ou les méthodes de calcul de racines de matrices (voir [Rob80], [KL95], [Hig97] et les références incluses).

Les algorithmes basés sur la fonction signe matricielle ont été largement utilisés pour la résolution de problèmes ARE dans le cas constant ; pour les systèmes linéaires continus [KLJ89], [KL92], [KL95], [Q-OH98], [Hig08] et discrets [FB99]. De récents travaux tentent d'étendre ces résultats au cas paramétrique [RV10], [GYC12]. Plus particulièrement, la méthode développée dans [RV10], consiste en l'utilisation des itérations de Newton associées aux transformations linéaires fractionnelles (LFT) pour le calcul de la fonction signe matricielle à dépendance paramétrique. Néanmoins, une réduction d'ordre est nécessaire à chaque étape. La méthode proposée dans cette section ne se base pas sur les itérations de Newton pour le calcul de la fonction signe matricielle mais sur des itérations n'employant que des multiplications et un calcul unique d'inverse. Ces itérations convergent vers une approximation rationnelle de la solution d'une équation de Riccati ou de Lyapunov. D'un côté, cela permet de revisiter, à travers une nouvelle approche, la résolution des équations de Riccati à l'aide de la fonction signe (ou racine) matricielle pour le cas non paramétré et ; d'un autre côté, cela permet de proposer une approche généralisable au cas multi-paramétrique.

Le chapitre présente en premier lieu l'application de la fonction signe pour la solution d'une équation de Riccati standard, que nous noterons désormais ARE, et le lien entre la racine carrée et une équation de Riccati particulière, notée CARE [I-CM-T77] (Particulière dans le sens où le système n'admet que des pôles réels et sous certaines contraintes supplémentaires). Ensuite, se basant sur de récents travaux concernant la caractérisation de matrices palindromiques quadratiques polynomiales de la fonction signe et des racines matricielles (voir [IM11]), une nouvelle approche est proposée utilisant les expansions en séries de Laurent de systèmes LTI. Puis, l'extension au cas paramétré de ces approches est également décrite.

4.1.2 Préliminaires

4.1.2.1 Rappel sur les itérations de Newton-Shultz (N-S)

On reprendra dans cette partie l'algorithme de N-S donnée par (3.13) et sa propriété de convergence :

Théorème 4.1 : (*Convergence des N-S itérations*)

Posons $Z \in \mathbb{C}^{n \times n}$ n'ayant aucune valeur propre sur l'axe des imaginaires.

$$\text{Si } \|I - Z^2\| < 1 \text{ alors } Z_k \rightarrow \text{sign}(Z) \text{ quand } k \rightarrow \infty \quad (4.1)$$

$$\text{Et } \|I - Z_k^2\| < \|I - Z^2\|^{2^k}.$$

Preuve : voir [KL92].

Remarque 4.1 : Le Théorème 4.1 décrit une convergence locale des N-S itérations.

Lemme 4.1 :

Posons $Z \in \mathbb{C}^{n \times n}$ une matrice complexe n'ayant aucune valeur propre sur l'axe des imaginaires.

$$\text{sign}(Z) = Z(Z^2)^{-1/2} \quad (4.2)$$

Preuve : Voir le Chapitre 2 sur la fonction signe matricielle.

Considérons maintenant une matrice $Z \in \mathbb{C}^{n \times n}$ n'ayant pas de valeurs propres à parties réelles nulles. Notons :

$$\text{sign}\left(\begin{bmatrix} 0 & Z \\ I_n & 0 \end{bmatrix}\right) = \begin{bmatrix} 0 & Z^{-1/2} \\ Z^{1/2} & 0 \end{bmatrix} \quad (4.3)$$

Le calcul de la racine matricielle s'effectue à l'aide de l'algorithme suivant :

$$\begin{cases} Y_0 = Z \text{ et } W_0 = I_n \\ Y_{k+1} = \frac{1}{2}Y_k(3I_n - W_k Y_k), \quad k = 1, 2, 3, \dots \\ W_{k+1} = \frac{1}{2}(3I_n - W_k Y_k)W_k \end{cases} \quad (4.4)$$

avec $\lim_{k \rightarrow \infty} Y_k = Z^{\frac{1}{2}}$ et $\lim_{k \rightarrow \infty} W_k = Z^{-\frac{1}{2}}$.

Théorème 4.2 : (Convergence des N-S itérations)

Posons $Z \in \mathbb{C}^{n \times n}$ n'ayant aucune valeur propre réelle non-négative.

$$\text{Si } \left\| \begin{bmatrix} I_n - Z & 0 \\ 0 & I_n - Z \end{bmatrix} \right\| < 1 \text{ alors } \begin{bmatrix} 0 & Y_k \\ W_k & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & Z^{\frac{1}{2}} \\ Z^{-\frac{1}{2}} & 0 \end{bmatrix} \text{ pour } k \rightarrow \infty \quad (4.5)$$

4.1.2.2 Application aux équations de Riccati

Deux différentes équations de Riccati sont considérées dans cette partie. La première est une ARE standard de la forme :

$$A^T X + XA - XBR^{-1}B^T X + C^T C = 0 \quad (4.6)$$

avec les matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ et $R \in \mathbb{R}^{m \times m}$ et les hypothèses standard :

(H1) R est définie positive,

(H2) (A, B) est stabilisable,

(H3) (A, C) est détectable.

Avec la matrice Hamiltonienne associée :

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \in \mathbb{R}^{2n \times 2n} \quad (4.7)$$

et en se basant sur les hypothèses (H2)-(H3), H n'a pas de valeurs propres sur l'axe des imaginaires.

Une sous classe de problèmes ARE est également considérée ici. Le problème, initialement introduit dans [Jon76] et étendu dans [I-CM-T77], concerne les systèmes avec pôles réels sous les hypothèses sur la matrice de sortie suivantes :

(H4) C est inversible,

(H5) $(C^T C).A$ est une matrice symétrique.

On notera ce problème (CARE). Se basant sur les résultats obtenus dans [I-CM-T77], la solution du problème CARE (4.6) est donnée par :

$$X = C^T \left(\left(CBR^{-1}B^T C^T + CA^2 C^{-1} \right)^{1/2} - CAC^{-1} \right)^{-1} C \quad (4.8)$$

Ce qui mène à un lien direct entre la solution de ce problème contraint et la racine matricielle définie positive de $CBR^{-1}B^T C^T + CA^2 C^{-1}$.

Remarque 4.2 : La solution d'un problème ARE standard implique le calcul d'une fonction signe matricielle de taille $2n \times 2n$ contrairement au problème CARE qui lui est lié à une racine matricielle d'ordre $n \times n$.

4.1.3 Problèmes ARE et CARE constants

La méthode proposée se base sur une matrice de Laurent palindromique polynomiale de la forme suivante :

$$L(z, P, Q) \triangleq Pz^{-1} + Q + Pz \quad (4.9)$$

avec $z \in \mathbb{C}$, $P \in \mathbb{R}^{n \times n}$ et $Q \in \mathbb{R}^{n \times n}$.

Lemme 4.2 : [IM11]

La matrice polynomiale de Laurent (4.9) est inversible dans un anneau contenant le cercle unité si et seulement si $M = Q^{-1}P$ n'a pas de valeurs propres réelles de module plus grand ou égal à $1/2$. De plus, cette matrice est inversible pour tout $z \in \mathbb{C}$ tel que $r < |z| < 1/r$, avec :

$$r = \rho \left(-2M \left(I + (I - 4M^2)^{1/2} \right)^{-1} \right) \quad (4.10)$$

et l'inverse $L(z, P, Q)^{-1} \triangleq \sum_{i=-\infty}^{i=+\infty} F_i z^i$ est telle que F_0 est donnée par :

$$F_0 = (I - 4M^2)^{-\frac{1}{2}} Q^{-1} \quad (4.11)$$

Preuve : voir Lemme 7 dans [IM11].

4.1.4 Un approximant rationnel de la fonction signe matricielle

Une des contributions de cette partie est de présenter une approximation rationnelle de la fonction signe matricielle se basant sur la nouvelle méthode du calcul de la matrice constante F_0 donnée par (4.11).

Théorème 4.3 : (une nouvelle caractérisation de la matrice signe)

Posons $Z \in \mathbb{R}^{n \times n}$ une matrice n'ayant pas de valeurs propres sur l'axe des imaginaires telle que $r = \rho \left(2(I_n + Z^2)^{-1} - I_n \right) < 1$, alors la fonction signe matricielle de Z est donnée par :

$$\text{sign}(Z) = 2(I_n + Z^2)^{-1} Z \cdot \left(I_n + \sum_{l=1}^{\infty} \binom{2l}{l} \frac{1}{2^{2l}} \left(2(I_n + Z^2)^{-1} - I_n \right)^{2l} \right) \quad (4.12)$$

Preuve : Considérons d'abord $L \left(z, \frac{1}{4}(Z^{-1} - Z), \frac{1}{2}(Z^{-1} + Z) \right)$ et le changement de variable suivant :

$$\tilde{z}^{-1} = z + \frac{1}{z}, \quad z \in \mathbb{C}, z \neq 0, r < |z| < 1/r \quad (4.13)$$

Notons que si $r < 1$ alors \tilde{z} existe. Ensuite, se basant sur (4.9) :

$$L \left(\tilde{z}, \frac{1}{4}(Z^{-1} - Z), \frac{1}{2}(Z^{-1} + Z) \right) = \frac{1}{4}(Z^{-1} - Z) \tilde{z}^{-1} + \frac{1}{2}(Z^{-1} + Z) \quad (4.14)$$

Puis, l'inverse de la matrice polynomiale de Laurent $L(\tilde{z}) \triangleq L\left(\tilde{z}, \frac{1}{4}(Z^{-1} - Z), \frac{1}{2}(Z^{-1} + Z)\right)$ peut être exprimé comme une fonction de transfert propre avec une représentation d'état de la forme :

$$(L(\tilde{z}))^{-1} = D_L + C_L (\tilde{z}I_n - A_L)^{-1} B_L \quad (4.15)$$

avec $A_L = B_L = -(I_n + Z^2)^{-1} + 0.5I_n$ et $C_L = D_L = 2(I_n + Z^2)^{-1} Z$. En utilisant l'expansion en séries de Laurent de (4.15) on obtient :

$$(L(\tilde{z}))^{-1} = \tilde{F}_0 + \sum_{k=1}^{\infty} \tilde{F}_k \tilde{z}^{-k} = \tilde{F}_0 + \sum_{k=1}^{\infty} \tilde{F}_k \left(z + \frac{1}{z}\right)^{-k} \quad (4.16)$$

où $\tilde{F}_0 = D_L$ et $\tilde{F}_k = C_L A_L^k B_L, k=0,1,\dots$ sont des paramètres de Markov. A l'aide du Théorème 8 dans [IM11], pour le choix particulier de la paire de matrices (P, Q) : $P = \frac{1}{4}(Z^{-1} - Z)$ et $Q = \frac{1}{2}(Z^{-1} + Z)$, la matrice constante F_0 donnée par (4.11) est le signe matriciel de Z . A partir de (4.16) les termes constants apparaissent pour les puissances paires de la série, d'où :

$$\text{sign}(Z) = F_0 = \tilde{F}_0 + \sum_{l=1}^{\infty} \binom{2l}{l} \tilde{F}_{2l} = D_L + \sum_{l=1}^{\infty} \binom{2l}{l} C_L A_L^{2l} B_L \quad (4.17)$$

Finalement, avec les expressions de A_L, B_L, C_L et D_L (4.15), (4.17) est équivalente à (4.12). ■

La série infinie (4.12) considérée sur les q premiers termes sera écrite :

$$Z_q = 2(I_n + Z^2)^{-1} Z \left(I_n + \sum_{l=1}^q \binom{2l}{l} \frac{1}{2^{2l}} \left(2(I + Z^2)^{-1} - I_n \right)^{2l} \right) \quad (4.18)$$

Z_q dénote donc l'approximation rationnelle de $\text{sign}(Z)$ au $q^{\text{ième}}$ terme.

Remarque 4.3 : Notons que le calcul de $\text{sign}(Z)$ à l'aide de (4.12) n'implique que l'inversion de la matrice constante $I_n + Z^2$.

Remarque 4.4 : on rappelle que la condition (4.1) est nécessaire pour la convergence des itérations de Newton-Shultz (N-S). L'intérêt de l'expansion aux polynômes de Laurent est de permettre d'atteindre la condition de convergence rapidement car n'utilisant qu'une unique inversion matricielle.

L'algorithme s'écrit en deux étapes « *Laurent* + *N-S* » :

Algorithme 4.1 : Soit une matrice $Z \in \mathbb{R}^{n \times n}$, sans valeur propre sur l'axe des imaginaires, telle que $\rho\left(2(I_n + Z^2)^{-1} - I_n\right) < 1$ et une tolérance ε pour le test de la convergence.

- 1 $X_0 = Z$, $k = 0$, $0 < \varepsilon \ll 1$.
- 2 Calcul de $(I_n + Z^2)^{-1}$
- 3 Tant que $\|I_n - X_k^2\| \geq 1$, (la condition de convergence de N-S n'est pas garantie)
- 4 $k = k + 1$, $X_k = Z_k$, (Z_k est donné par (4.18))
- 5 Fin (algorithme « *Laurent* »)
- 6 Tant que $(\|X_k - X_{k-1}\| / \|X_k\|) > \varepsilon$
- 7 $X_{k+1} = \frac{1}{2} X_k (3I_n - X_k^2)$, $k = k + 1$
- 8 Fin (algorithme « *N-S* »)
- 9 $\text{sign}(Z) = X_k$.

Notons qu'une méthode de *scaling* (ou de mise à l'échelle, voir chapitre 2) est possible et peut être rajoutée à l'Algorithme 1. Dans ce cas, un test de convergence plus approprié pour les itérations de N-S peut être utilisé à la place de celui donné en ligne 6 de ce même algorithme.

Remarque 4.5 : Dans le cas scalaire la condition de convergence pour N-S à partir des polynômes de Laurent est atteinte en une itération. Effectivement, appliquons (4.18) au cas scalaire $z \in \mathbb{C}$ avec $q = 1$, ce qui s'écrit :

$$z_1 = \frac{2z}{1+z^2} \left(1 + \frac{1}{4} \left(\frac{2}{1} \right) \left(\frac{1-z^2}{1+z^2} \right)^2 \right) = \frac{z(3z^4 + 2z^2 + 3)}{(z^2 + 1)^3}$$

$\forall z$ vérifiant $\rho(r) < 1$, z_1 vérifie $|1 - z_1^2| < 1$.

A la suite de la remarque précédente, il est important de noter qu'en général une faible valeur q de la matrice d'approximation rationnelle Z_q (4.18) est suffisante pour que la condition $\|I_n - X_k^2\| < 1$ soit vérifiée. Ce qui permet de passer aux itérations de N-S rapidement. Cette observation a été faite sur un grand nombre d'exemples tels que ceux décrits dans l'Exemple 4.1 et l'Exemple 4.2.

A titre d'illustration, la Fig. 4.1 présente deux zones de scalaires z (bleue et rouge), vérifiant $r < |z| < \frac{1}{r}$ et telles que la condition (4.10) soit vérifiée, leur polynôme de Laurent associé est donc inversible. La zone bleue correspond aux points qui vérifient la condition (4.1) de convergence des itérations de Newton-Shultz (N-S), i.e. pour lesquels N-S est applicable directement, la zone rouge est la zone où l'algorithme proposé est applicable. Une illustration semblable peut-être réalisée dans le cas matriciel en s'intéressant à une des valeurs propres de Z .

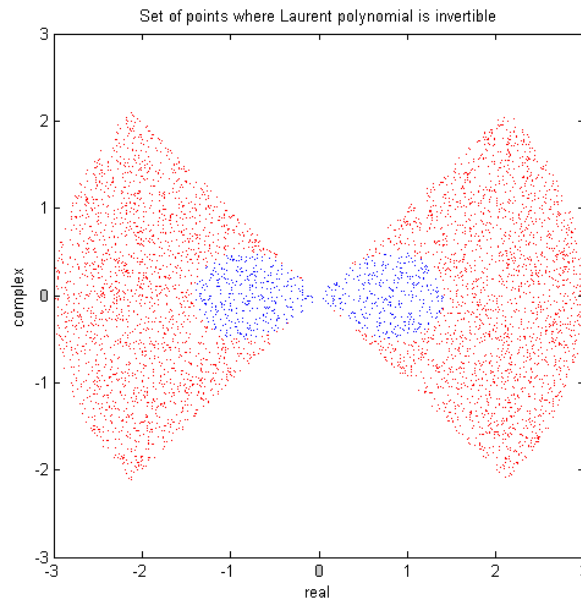


Fig. 4.1 : Points scalaires vérifiant la condition (4.10)

Quelques exemples d'utilisation de l'algorithme sont présentés. Tous les exemples testés dans l'Exemple 4.1 et l'Exemple 4.2 vérifient la condition $\rho\left(2\left(I_n + Z^2\right)^{-1} - I_n\right) < 1$ et sont comparés par rapport à l'algorithme de Newton.

Exemple 4.1 : Cet exemple est repris de [KBS94] et utilise des matrices proposées, par exemple dans [Hig91]. Soit $Z = UDU^T$, où $D = \text{randsvd}(n, \kappa, 3, kl, ku)$; on rappelle que “randsvd” est une fonction qui permet d'obtenir une matrice aléatoire avec des valeurs singulières prédéfinies avec : n la dimension (ici 4, 8 ou 16), $\kappa=100$ le conditionnement de la matrice (plus κ est élevé plus la matrice est mal conditionnée) et kl (resp. ku) représente le nombre de sous-diagonales (resp. sur-diagonales) non nuls (ici 2, 4 ou 8) et le mode étant choisi à 3 les valeurs singulières sont géométriquement distribuées. Une tolérance $\varepsilon=10^{-10}$ est considérée. La Table 4.1 résume les résultats. Chaque case représente un résultat « moyen » obtenu pour différentes tailles de matrices.

	Algorithme 4.1	Newton
--	----------------	--------

n	Nombre q pour l'approximation rationnelle Z_q	Nombre d'itérations N-S	Nombre d'itérations de Newton
4	1	5	6
8	2	5	7
16	2	6	9

Table 4.1 : Résultats pour l'Exemple 4.1.

Exemple 4.2 : Cet exemple est emprunté à [KL92]. Soit $Z = \text{diag}(\lambda_1, \dots, \lambda_n) + T \in \mathbb{R}^{n \times n}$ avec λ_i aléatoirement et uniformément distribué dans $[-10, 10]$ et T une matrice triangulaire supérieure avec des valeurs uniformément distribuées dans $[-1, 1]$. La tolérance choisie est $\varepsilon = 10^{-10}$. La Table 4.2 résume les résultats obtenus. Chaque case représente un résultat « moyen » obtenu pour différentes tailles de matrices.

n	Algorithme 4.1		Newton
	Nombre q pour l'approximation rationnelle Z_q	Nombre d'itérations N-S	Nombre d'itérations de Newton
4	1	5	7
8	1	6	8
16	2	6	9

Table 4.2 : Résultats pour l'Exemple 4.2.

Ces exemples montrent que l'Algorithme 4.1 mène à une solution avec une certaine précision, avec le même nombre d'itérations approximativement que pour la méthode de Newton et n'utilise qu'une seule inversion matricielle.

Exemple 4.3 : Considérons le problème ARE de la forme (4.6) avec les données suivantes :

$$A = \begin{bmatrix} 3 & 1 & 4 \\ -1 & 2 & 5 \\ -1 & 3 & -2 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, C = [1 \quad 2 \quad 0], R = 1$$

La matrice Hamiltonienne associée H n'a pas de valeurs propres sur l'axe des imaginaires et vérifie $\rho\left(2(I_6 + H^2)^{-1} - I_6\right) < 1$.

Appliquant l'Algorithme 4.1, la première approximation rationnelle (donnée par (4.18) avec $q=1$) de $\text{sign}(H)$ est :

$$Z_1 = \begin{bmatrix} 0.720 & -0.479 & 0.135 & -0.017 & -0.037 & 0.022 \\ 0.101 & 0.162 & 0.578 & -0.037 & -0.150 & 0.053 \\ 0.083 & 0.406 & -0.193 & 0.022 & 0.053 & -0.029 \\ -0.450 & 0.065 & -0.493 & -0.720 & -0.101 & -0.083 \\ 0.065 & -0.010 & 0.089 & 0.479 & -0.162 & -0.406 \\ -0.493 & 0.089 & -0.766 & -0.135 & -0.578 & 0.193 \end{bmatrix}$$

et vérifie $\|I_6 - Z_1^2\|_2 = 0.989 < 1$. 8 itérations des étapes N-S (ligne 6 de l'Algorithme 4.1) donnent :

$$\text{sign}(H) = \begin{bmatrix} 0.996 & -0.313 & 0.566 & -0.032 & -0.053 & 0.038 \\ -0.008 & 0.390 & 1.083 & -0.053 & -0.148 & 0.049 \\ -0.026 & 0.725 & -0.370 & 0.038 & 0.049 & -0.131 \\ -0.532 & -0.210 & -0.402 & -0.996 & 0.008 & 0.026 \\ -0.210 & -0.342 & -0.013 & 0.313 & -0.390 & -0.725 \\ -0.402 & -0.013 & -0.836 & -0.566 & -1.083 & 0.370 \end{bmatrix}$$

avec tolérance de convergence prédéfinie $\varepsilon = 10^{-10}$. Finalement, la solution positive approximée du problème ARE donné est :

$$\tilde{X} = \begin{bmatrix} 207.31 & -63.151 & 36.043 \\ -63.151 & 31.969 & -0.817 \\ 36.043 & -0.817 & 14.857 \end{bmatrix}$$

4.1.5 Méthode particulière pour le calcul de racines matricielles

L'approche décrite permet également une nouvelle approximation rationnelle de la racine matricielle d'une matrice donnée Z .

Théorème 4.4 : *(Une nouvelle caractérisation de la racine matricielle)*

Soit $Z \in \mathbb{R}^{n \times n}$ une matrice n'ayant pas de valeur propre non positive telle que $\rho(2Z(I_n + Z)^{-1} - I_n) < 1$, alors la racine matricielle de Z est donnée par :

$$Z^{1/2} = 2 \left(I_n + \sum_{l=1}^{\infty} \binom{2l}{l} \frac{1}{2^{2l}} \left(2Z(I_n + Z)^{-1} - I_n \right)^{2l} \right) Z(I_n + Z)^{-1} \quad (4.19)$$

Preuve : Elle se fait de manière similaire à la preuve du Théorème 4.3 en utilisant le polynôme de Laurent $L\left(\tilde{z}, \frac{1}{4}(I_n - Z^{-1}), \frac{1}{2}(I_n + Z^{-1})\right)$ et les matrices suivantes

$$A_L = C_L = (I_n + Z^{-1})^{-1} - 0.5I_n \text{ et } B_L = D_L = 2(I_n + Z^{-1})^{-1}. \quad \blacksquare$$

On définit également $Z_q^{1/2}$ l'approximation au $q^{ième}$ terme de $Z^{1/2}$:

$$Z_q^{1/2} = 2 \left(I_n + \sum_{l=1}^q \binom{2l}{l} \frac{1}{2^{2l}} (2Z(I_n + Z)^{-1} - I_n)^{2l} \right) Z(I_n + Z)^{-1} \quad (4.20)$$

Algorithme 4.2 : Soit une matrice $Z \in \mathbb{R}^{n \times n}$ n'ayant pas de valeurs propres non positives telle que $\rho(2Z(I_n + Z)^{-1} - I_n) < 1$ et une tolérance ε pour tester la convergence.

- 1 $X_0 = Z, W_0 = I_n, k = 0, 0 < \varepsilon < 1.$
- 2 Calcul de $(I_n + Z)^{-1}$
- 3 Tant que $\left\| \begin{bmatrix} X_k - I_n & 0 \\ 0 & X_k - I_n \end{bmatrix} \right\| \geq 1$ (condition de convergence de N-S)
- 4 $k = k + 1, X_k = Z_k^{1/2},$ ($Z_k^{1/2}$ est donnée par (4.20))
- 5 fin
- 6 Tant que $(\|X_k - X_{k-1}\| / \|X_k\|) > \varepsilon$
- 7
$$\begin{cases} X_{k+1} = \frac{1}{2} X_k (3I_n - W_k X_k) \\ W_{k+1} = \frac{1}{2} (3I_n - W_k X_k) W_k \end{cases}, W_k = X_k X_0^{-1}, k = k + 1$$
- 8 fin
- 9 $Z^{1/2} = X_k.$

Exemple 4.4 : Considérons le problème CARE avec les données suivantes :

$$A = \begin{bmatrix} 2 & -1 & 5 \\ 1 & 5 & -3 \\ 2 & -1 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R = 1 \quad (4.21)$$

Il est aisé de voir que ces matrices vérifient les hypothèses (H3)-(H5) de la section 4.1.1.2. Appliquant l'Algorithme 4.2, la première approximation rationnelle (donnée par (4.20) avec $q=1$) de $(CBR^{-1}B^T C^T + CA^2 C^{-1})^{1/2}$ est :

$$Z_1^{1/2} = \begin{bmatrix} 2.484 & 0.131 & 0.247 \\ 0.131 & 2.587 & -0.348 \\ 0.247 & -0.348 & 2.262 \end{bmatrix}$$

avec $\|Z_1^{1/2} - I_3\|_2 = 0.773 < 1$. Donc, la condition pour le passage aux itérations N-S est vérifiée. Après 8 itérations N-S, on trouve :

$$X_8 = \begin{bmatrix} 3.654 & 0.261 & 0.763 \\ 0.261 & 4.837 & -1.594 \\ 0.763 & -1.594 & 3.447 \end{bmatrix}$$

avec la tolérance de convergence $\varepsilon = 10^{-10}$. Ensuite, la solution approximée du problème CARE est :

$$\tilde{X} = C^T (X_8 - CAC^{-1})^{-1} C = \begin{bmatrix} 1850.5 & 3686.2 & -119.57 \\ 3686.2 & 7378.9 & -258.72 \\ -119.57 & -258.72 & 19.937 \end{bmatrix}$$

4.1.6 Extension au cas paramétré

Dans cette section, les matrices des problèmes ARE et CARE sont supposées appartenir à la classe suivante de matrices paramétrées :

$$Z(\theta) \triangleq \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n} \quad (4.22)$$

avec $\bar{Z} \in \mathbb{R}^{n \times n w^\gamma}$, $\bar{\theta}_k = [\theta_k^{-g} \ \cdots \ \theta_k^{-1} \ 1 \ \theta_k^1 \ \cdots \ \theta_k^h]^T \in \mathbb{R}^w$, $\forall k \in [1, \gamma]$ et $\bar{\theta} = [\bar{\theta}_1 \otimes \cdots \otimes \bar{\theta}_\gamma]^T \in \mathbb{R}^{w^\gamma}$. $Z(\theta)$ est supposée non singulière pour tout θ . L'arithmétique sur cette classe reste la même que pour les classes définies dans les chapitres précédents.

Remarque 4.6 : Pour le problème standard ARE les hypothèses (H1)-(H3) et la condition $\rho\left(2\left(I_n + H(\theta)^2\right)^{-1} - I_n\right) < 1$, où $H(\theta)$ est la matrice Hamiltonienne, sont supposées vraies dans une boule de norme l_2 de rayon δ et centrée en θ_0 . De manière similaire, les hypothèses (H1)-(H5) et la condition $\rho\left(2Z(\theta)(I_n + Z(\theta))^{-1} - I_n\right) < 1$, où

$$Z(\theta) = C(\theta)B(\theta)R(\theta)^{-1}B(\theta)^T C(\theta)^T + C(\theta)A(\theta)^2 C(\theta)^{-1}$$

sont supposées vraies dans une boule de norme l_2 de rayon δ et centrée en θ_0 .

Exemple 4.5 : Considérons le problème ARE avec les matrices paramétrées suivantes :

$$A(\theta) = \begin{bmatrix} \theta-3 & 1-2\theta^2 \\ -1 & -3\theta \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C(\theta) = [\theta \quad -1], R = 1$$

avec $\theta_0 = 0$ et $\delta = 1.5$. La matrice Hamiltonienne associée s'écrit :

$$H(\theta) = \begin{bmatrix} \theta-3 & 1-2\theta^2 & 0 & 0 \\ -1 & -3\theta & 0 & -1 \\ -\theta^2 & \theta & 3-\theta & 1 \\ \theta & -1 & 2\theta^2-1 & 3\theta \end{bmatrix}$$

L'intérêt de la méthode développée est que son passage au paramétrique se fait avec les mêmes algorithmes que dans le cas classique en rajoutant les opérations arithmétiques définies lemme 4.1 (3.81)~(3.83). Pour illustrer cette remarque, on utilise l'algorithme 4.1 avec l'approximation rationnelle au 1^{er} ordre ($q=1$ dans (4.18)) permettant de vérifier la condition de convergence de N-S et pour la 2^{ème} partie de l'algorithme un nombre d'itérations fixé *a priori* à $n = 6$. Ce nombre est un bon compromis entre précision de la solution et convergence de l'algorithme. Les figures Fig. 4.2 et Fig. 4.3 montrent les résultats de l'algorithme. La première partie « *Laurent* » Fig. 4.2 montre que la condition $\|I_4 - (H_1(\theta))^2\|_2 < 1$ est valable pour tout $\theta \in [-1.5, 1.5]$ où $H_1(\theta)$ est la première approximation rationnelle donnée par (4.18).

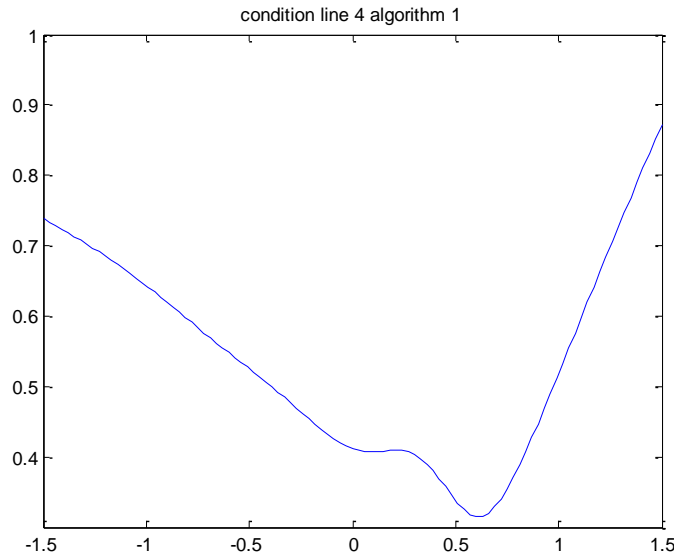


Fig. 4.2 : La vérification de la condition de N-S pour la première approximation rationnelle de la fonction signe.

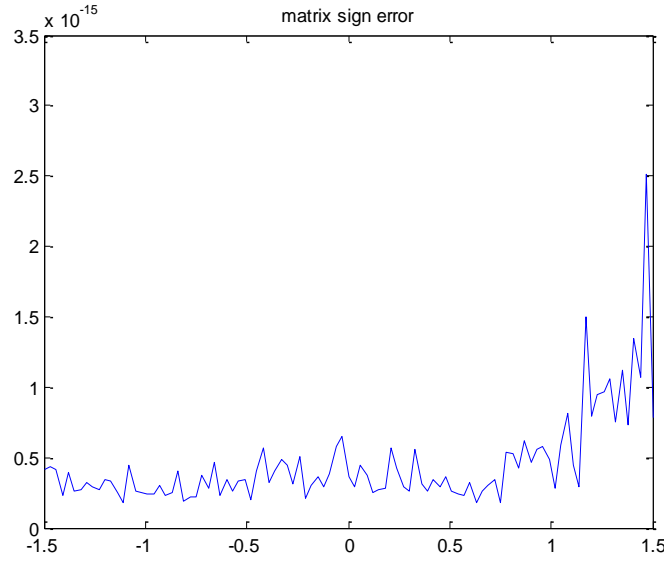


Fig. 4.3 : L'erreur $\|\tilde{S}(\theta_i) - S_N(\theta_i)\|_2$.

Afin de comparer les résultats sur une base connue, la méthode des itérations de Newton a été calculée pour déterminer $S_N(\theta)$ pour de nombreux points dans $[-1.5, 1.5]$. La Fig. 4.3 illustre l'erreur faite sur ces points entre Newton et l'algorithme 4.1 $\tilde{S}(\theta_i)$, i.e. $\|\tilde{S}(\theta_i) - S_N(\theta_i)\|_2$, $\theta_i \in [-1.5, 1.5]$. Elle montre la pertinence de l'approche.

Enfin, la solution, matrice approximée $\tilde{X}(\theta)$, qui satisfait le problème ARE est obtenue avec une norme de l'erreur résiduelle maximum de :

$$\max_{\theta \in [-\delta, \delta]} \|A(\theta)^T \tilde{X}(\theta) + \tilde{X}(\theta) A(\theta) - \tilde{X}(\theta) B R^{-1} B^T \tilde{X}(\theta) + C(\theta)^T C(\theta)\| \approx 5.69 \cdot 10^{-14}$$

Exemple 4.6 : Après avoir illustré le cas ARE, considérons le problème CARE avec les matrices paramétrées suivantes :

$$A(\theta) = \begin{bmatrix} 4\theta + 3 & 1 - \theta & 3\theta + 1 \\ 2 & \theta^2 + 1 & \theta + 3 \\ -4\theta - 2 & \theta + 1 & -7\theta - 2 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, C(\theta) = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R = 1$$

avec $\theta_0 = 0$ et $\delta = 1.5$. L'Algorithme 4.2 est appliqué pour trouver la racine matricielle de

$$\begin{aligned}
 Z(\theta) &= C(\theta)BR^{-1}B^TC(\theta)^T + C(\theta)A(\theta)^2C(\theta)^{-1} = \\
 &= \begin{bmatrix} (4\theta+2)^2+6 & -2\theta^2-6\theta+2 & 12\theta^2+4\theta+1 \\ -2\theta^2-6\theta+2 & (\theta+1)^2+(\theta^2+1)^2+4 & \theta^3-2\theta^2-10\theta-3 \\ 12\theta^2+4\theta+1 & \theta^3-2\theta^2-10\theta-3 & 26\theta^2+18\theta+6 \end{bmatrix} \quad (4.23)
 \end{aligned}$$

De la même manière que dans l'exemple précédent les choix pour l'algorithme 4.2 sont $q=1$ pour l'approximation rationnelle et $n=8$ pour le nombre d'itérations de N-S. Les Fig. 4.4 et Fig. 4.5 présentent les résultats. La Fig. 4.4 montre que la condition de convergence de N-S est bien respectée, i.e. $\|Z_1^{1/2}(\theta) - I_3\|_2 < 1$ pour tout $\theta \in [-\delta, \delta]$ où $Z_1^{1/2}(\theta)$ est la première approximation rationnelle donnée par (4.20).

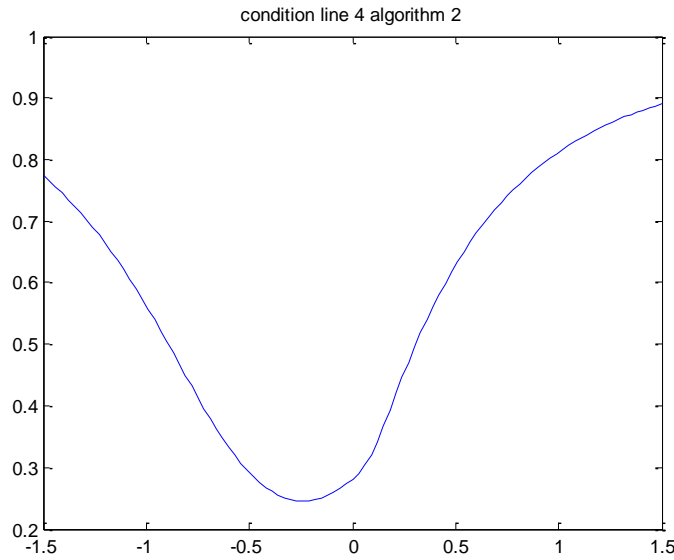


Fig. 4.4 : La vérification de la condition N-S pour la première approximation rationnelle de la racine matricielle.

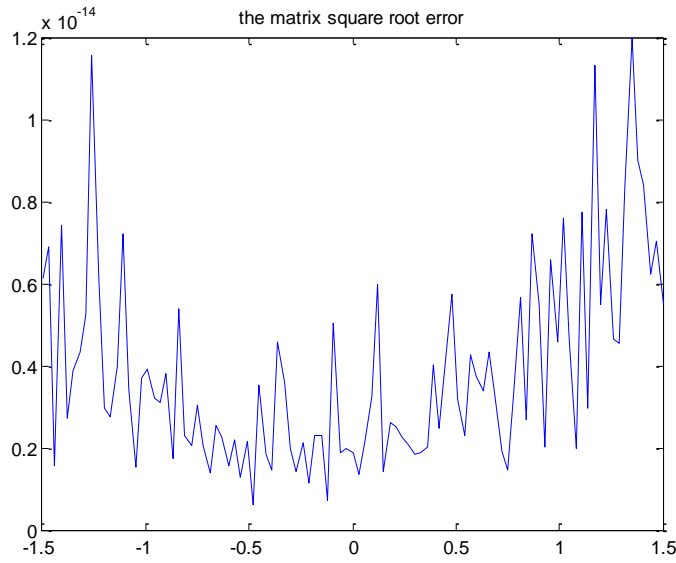


Fig. 4.5 : L'erreur $\|\tilde{S}^{1/2}(\theta_i) - S_N^{1/2}(\theta_i)\|_2$, $\theta_i \in [-1.5, 1.5]$

Comme pour l'Exemple 4.5, afin de comparer les résultats sur une base connue, la méthode des itérations de Newton a été calculée pour déterminer $S_N^{1/2}(\theta_i)$ pour de nombreux points dans $[-1.5, 1.5]$. La Fig. 4.5 illustre l'erreur faite sur ces points entre Newton et l'algorithme 4.1 $\tilde{S}^{1/2}(\theta_i) \tilde{S}(\theta_i)$, i.e. $\|\tilde{S}^{1/2}(\theta_i) - S_N^{1/2}(\theta_i)\|_2$, $\theta_i \in [-1.5, 1.5]$ et montre à nouveau la pertinence de l'approche.

Enfin, la solution, matrice approximée $\tilde{X}(\theta)$, qui satisfait le problème CARE avec une norme de l'erreur résiduelle maximum de :

$$\max_{\theta \in [-\delta, \delta]} \|A(\theta)^T \tilde{X}(\theta) + \tilde{X}(\theta) A(\theta) - \tilde{X}(\theta) B R^{-1} B^T \tilde{X}(\theta) + C(\theta)^T C(\theta)\| \approx 2.5 \cdot 10^{-7}$$

En conclusion, lorsque la condition sur les polynômes de Laurent est vérifiée, théorème 4.3, la méthode présente de réels intérêts et notamment pour travailler avec des systèmes d'ordres élevés. L'avantage résidant dans le fait de n'utiliser qu'une seule inversion matricielle pour tout l'algorithme. Néanmoins, la méthode reste limitée à cette classe de matrice particulière et il serait intéressant de voir si des extensions sont possibles à d'autres types de matrices. Ces travaux ont fait office d'une présentation en conférence [GYC14b]. Une autre méthode d'itération a été développée permettant de traiter un ensemble de matrices considérablement plus large et présentée dans la section suivante.

4.2 Combinaison de la fonction signe matricielle et de l'inversion DFT

4.2.1 Introduction

Dans cette partie, une méthode itérative de résolution d'équations de Riccati paramétrées est présentée basée sur la fonction signe matricielle. Cette méthode repose sur deux résultats : une expansion fractionnaire de l'approximation rationnelle de la fonction signe matricielle via Padé d'un côté, et une transformation de Fourier discrète (DFT) basée sur l'inversion de matrices à dépendance paramétrique de l'autre. Dans cette partie, une application au cas d'extractions d'harmoniques d'ondes de courant ou de tension illustre la validité de cette approche.

4.2.2 Classe des matrices

La classe de matrices considérée ici ne prend en compte que des paramètres à puissances positives, c.à.d. une dépendance matricielle polynomiale :

$$Z(\theta) \triangleq \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n} \quad (4.24)$$

avec $\theta = [\theta_1 \ \cdots \ \theta_p]^T$, $\bar{Z} \in \mathbb{R}^{n \times n\bar{m}}$, $\bar{\theta} = [\bar{\theta}_1 \otimes \cdots \otimes \bar{\theta}_p] \in \mathbb{R}^{\bar{m}}$, $\prod_{k=1}^{k=p} m_k = \bar{m}$ et $\bar{\theta}_k = [\theta_k^0 \ \theta_k^1 \ \cdots \ \theta_k^{m_k-1}]^T \in \mathbb{R}^{m_k}$, $k \in \{1, \dots, p\}$. Une autre notation $\bar{\theta} \triangleq \theta^{[m_1, \dots, m_p]}$ est aussi utilisée dans la suite. $Z(\theta)$ est supposée régulière, c.-à-d. $\det(Z(\theta)) \neq 0$. L'écriture (4.24) de $Z(\theta)$ décompose la classe de matrices en une matrice constante \bar{Z} (composée de \bar{m} matrices constantes de taille $n \times n$) et un vecteur de paramètres $\bar{\theta}$. Des opérations peuvent être définies sur cette classe de matrices et sont rappelées dans le lemme suivant.

Lemme 4.3: Soit $Z(\theta) = \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$, $V(\theta) = \bar{V}(\bar{\theta}' \otimes I_n) \in \mathbb{R}^{n \times n}$ avec $\bar{\theta} = \theta^{[m_1, \dots, m_p]}$ $\bar{\theta}' = \theta^{[q_1, \dots, q_p]}$, $\prod_{k=1}^{k=p} m_k = \bar{m}$ et $\alpha \in \mathbb{R}$ alors:

$$Z(\theta) + V(\theta) = (\bar{Z} + \bar{V})(\bar{\varphi} \otimes I_n) \quad (4.25)$$

avec : $\bar{\varphi} = \theta^{[\max(m_1, q_1), \dots, \max(m_p, q_p)]}$.

$$Z(\theta) \cdot V(\theta) = \bar{Z}(I_{\bar{m}} \otimes \bar{V})((\bar{\theta} \otimes \bar{\theta}') \otimes I_n) \quad (4.26)$$

$$\alpha I_n + Z(\theta) = (\alpha \bar{I} + \bar{Z})(\bar{\theta} \otimes I_n) \quad (4.27)$$

avec $\bar{I} = \begin{bmatrix} I_n & 0_{n \times (n-1)\bar{m}} \end{bmatrix}$.

Remarque 4.7 : Une réalisation minimale pour les matrices de la forme (4.24) peut être établie facilement. Par exemple, avec $\bar{Z} = [Z_1 \ \cdots \ Z_m]$, si un des termes est absent, i.e. $Z_i = 0_n$, la ligne correspondante peut être supprimée ainsi que la variable $\bar{\theta}_i$ associé. Si des doublons apparaissent dans $\bar{\theta}$ (par exemple : $\theta_i = \theta_j$ pour $i \neq j$), ils peuvent être rassemblés en sommant leurs matrices correspondantes Z_i, Z_j .

Exemple 4.7 : Considérons les matrices paramétrées suivantes :

$$Z(\theta) \triangleq \begin{bmatrix} -\theta_1 & 1 \\ 0 & -1 \end{bmatrix}, V(\theta) \triangleq \begin{bmatrix} 1 & \theta_2 \\ 2\theta_1^2 & \theta_1\theta_2 \end{bmatrix}, \theta_1, \theta_2 \in \mathbb{R} \quad (4.28)$$

la représentation de la forme (4.24) pour $Z(\theta)$ et $V(\theta)$ est donnée par :

$$Z(\theta) = \begin{bmatrix} 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta_1 \end{bmatrix} \otimes I_2 \right) \quad (4.29)$$

$$V(\theta) = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta_1 \\ \theta_1^2 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \theta_2 \end{bmatrix} \otimes I_2 \right) \quad (4.30)$$

Le Lemme 4.3 permet d'obtenir :

$$Z(\theta) + V(\theta) = \begin{bmatrix} 1 & 1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta_1 \\ \theta_1^2 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \theta_2 \end{bmatrix} \otimes I_2 \right) \quad (4.31)$$

$V(\theta)$ peut être réécrit, dans une forme minimale comme :

$$V(\theta) = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta_2 \\ \theta_1\theta_2 \\ \theta_1^2 \end{bmatrix} \otimes I_2 \right) \quad (4.32)$$

Ensuite, le produit $Z(\theta) \cdot V(\theta)$ donne :

$$\begin{aligned}
 Z(\theta) \cdot V(\theta) &= \\
 &= \begin{bmatrix} 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \left(I_2 \otimes \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \end{bmatrix} \right) \left(\begin{bmatrix} 1 \\ \theta_1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \theta_2 \\ \theta_1 \theta_2 \\ \theta_1^2 \end{bmatrix} \otimes I_2 \right) \\
 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ \theta_1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \theta_2 \\ \theta_1 \theta_2 \\ \theta_1^2 \end{bmatrix} \otimes I_2 \right)
 \end{aligned} \tag{4.33}$$

Et la forme minimale du produit s'écrit :

$$\begin{aligned}
 Z(\theta) \cdot V(\theta) &= \begin{bmatrix} 0 & 1 & 2 & 0 & -1 & 0 & 0 & -1 \\ 0 & -1 & -2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} \theta_1 \theta_2 \\ \theta_1^2 \\ \theta_1 \\ \theta_1 \theta_2 \end{bmatrix} \otimes I_2 \right) \\
 &= \begin{bmatrix} 0 & 0 & 2 & 0 & -1 & 0 \\ 0 & -1 & -2 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} \theta_1 \theta_2 \\ \theta_1^2 \\ \theta_1 \end{bmatrix} \otimes I_2 \right)
 \end{aligned} \tag{4.34}$$

Pour illustrer la propriété (4.27), on a :

$$\alpha I_n + Z(\theta) = (\alpha \bar{I} + \bar{Z}) (\bar{\theta} \otimes I_n) = \begin{bmatrix} \alpha & 1 & -1 & 0 \\ 0 & \alpha - 1 & 0 & 0 \end{bmatrix} (\bar{\theta} \otimes I_n) \tag{4.35}$$

4.2.3 Inversion de matrices polynomiales par la méthode de Transformation de Fourier Discrète

La définition d'une paire de matrices dite DFT a déjà été présentée dans la section 3.1.2.2, elle est reprise ici par souci de lisibilité.

Définition 4.1 : Les deux matrices $F(l)$ et $G(r)$ de $l, r = 0, 1, \dots, m-1$ sont dites paires DFT si les égalités suivantes sont satisfaites :

$$F(l) = \bar{G}(\rho_l \otimes I_n) \in \mathbb{C}^{n \times n} \tag{4.36}$$

$$G(r) = (N^{-1}) \bar{F}(\delta_r \otimes I_n) \in \mathbb{C}^{n \times n} \tag{4.37}$$

avec $\bar{F} = [F(0) \ \cdots \ F(m-1)]$, $\bar{G} = [G(0) \ \cdots \ G(m-1)]$, $\rho = \delta^{-1} = e^{\left(\frac{-2\pi j}{m}\right)}$,
 $\rho_l = [\rho^0 \ \rho^l \ \cdots \ \rho^{l(m-1)}]^T$, $\delta_r = [\delta^0 \ \delta^r \ \cdots \ \delta^{r(m-1)}]^T$.

Une extension de la Définition 4.1 au cas de plusieurs variables discrètes $l_k, r_k = 0, 1, \dots, N_k - 1$ ($k = 1, 2, \dots, p$) est donnée dans la Définition 4.2.

Définition 4.2: Les deux matrices $F(l_1, \dots, l_p)$ et $G(r_1, \dots, r_p)$ sont dites paires DFT si les égalités suivantes sont vérifiées :

$$F(l_1, \dots, l_p) = \bar{G}(\bar{\rho} \otimes I_n) \in \mathbb{C}^{n \times n}, \quad G(r_1, \dots, r_p) = \left(\prod_{k=1}^p N_k^{-1} \right) \bar{F}(\bar{\delta} \otimes I_n) \in \mathbb{C}^{n \times n} \quad (4.38)$$

avec $\bar{F} = [F(l_1, \dots, l_p)]_{l_k=[0, N_k-1] \ (k \in [1, p])}$, $\bar{G} = [G(r_1, \dots, r_p)]_{r_k=[0, N_k-1] \ (k \in [1, p])}$, $\rho_k = \delta_k^{-1} = e^{\left(\frac{-2\pi j}{N_k}\right)}$,
 $\bar{\rho} = [\bar{\rho}_1 \otimes \cdots \otimes \bar{\rho}_p]$, $\bar{\rho}_k = [\rho_k^0 \ \rho_k^{l_k} \ \cdots \ \rho_k^{l_k(N_k-1)}]^T$, $\bar{\delta} = [\bar{\delta}_1 \otimes \cdots \otimes \bar{\delta}_p]$,
 $\bar{\delta}_k = [\delta_k^0 \ \delta_k^{r_k} \ \cdots \ \delta_k^{r_k(N_k-1)}]^T$.

En prenant appui sur les travaux de [Yeu84] et [YK88], une méthode basée sur les DFT est utilisée dans la suite pour calculer l'inverse exact d'une matrice à dépendance polynomiale de la forme (4.24).

On considère une matrice multi-paramétrique $Z(\theta) \triangleq \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ de la forme (4.24)

et on pose $Z(\theta)^{-1} = \frac{1}{\det(Z(\theta))} \text{adj}(Z(\theta))$. Le polynôme $\det(Z(\theta))$ et la matrice à dépendance polynomiale $\text{adj}(Z(\theta))$ sont calculés à l'aide du Lemme 4.4, donné après. Seul le calcul de la matrice $\text{adj}(Z(\theta))$ sera présenté.

Lemme 4.4 : soit $Z(\theta)$ de la forme (4.24) avec $\bar{\theta} \triangleq \theta^{[m_1, \dots, m_p]}$ et $\prod_{k=1}^{k=p} m_k = \bar{m}$. La matrice $\text{adj}(Z(\theta))$ est donnée par :

$$\text{adj}(Z(\theta)) = \bar{Z}_a(\bar{\theta} \otimes I_n) = \underbrace{\left(\left(\frac{1}{\bar{m}} \right) \bar{Z}_\rho(\bar{\delta} \otimes I_n) \right)}_{\bar{Z}_a} (\bar{\theta} \otimes I_n) \quad (4.39)$$

$$\text{avec } \bar{Z}_\rho = [Z_\rho(l_1, \dots, l_p)]_{l_k \in [0, m_k] \text{ } (k \in [1, p])}, \quad Z_\rho(l_1, \dots, l_p) = \text{adj}(Z(\rho)), \quad \rho = [\rho_1 \quad \dots \quad \rho_p]^T, \\ \rho_k = e^{\left(\frac{-2\pi j}{m_k}\right)}, \quad \bar{\delta} = [\bar{\delta}_1 \otimes \dots \otimes \bar{\delta}_p], \quad \bar{\delta}_k = [\delta_k^0 \quad \delta_k^{r_k} \quad \dots \quad \delta_k^{r_k(m_k-1)}]^T \text{ et } \delta_k = e^{\left(\frac{2\pi j}{m_k}\right)}.$$

Preuve : Définissons le changement de variable suivant :

$$\theta_k = \rho_k^{l_k}, \quad k = 1, 2, \dots, p \quad (4.40)$$

En combinant (4.39) et (4.40) nous obtenons:

$$Z_a(\rho_1^{l_1}, \rho_2^{l_2}, \dots, \rho_p^{l_p}) = Z_\rho(l_1, l_2, \dots, l_p) \quad (4.41)$$

Ensuite, en calculant la fonction numérique $Z_\rho(l_1, l_2, \dots, l_p)$ et en prenant l'inverse DFT suivante:

$$\bar{Z}_a(r_1, \dots, r_p) = \left(\prod_{k=1}^p m_k^{-1} \right) \bar{Z}_\rho(\bar{\delta} \otimes I_n) = \left(\frac{1}{\bar{m}} \right) \bar{Z}_\rho(\bar{\delta} \otimes I_n) \quad (4.42)$$

on obtient la matrice \bar{Z}_a désirée. ■

Remarque 4.8 : La même procédure est appliquée pour le calcul de $\det(Z(\theta))$. A la place des puissances m_k , la somme des colonnes de plus haut degré pour chaque variable est considérée.

Exemple 4.8 : Considérons la matrice :

$$Z(\theta) = \begin{bmatrix} \theta_1 \theta_2 + 3 & -7\theta_1 \\ \theta_1 - \theta_2 & \theta_2 \end{bmatrix} = \bar{Z}(\bar{\theta} \otimes I_2)$$

$$\text{avec } \bar{Z} = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & -7 & 1 & 0 \\ 0 & 0 & -1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{\theta} = \begin{pmatrix} 1 \\ \theta_1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \theta_2 \end{pmatrix}.$$

Pour cet exemple nous avons $m_1 = 2$, $m_2 = 2$, $l_1 \in \{0, 1\}$, $l_2 \in \{0, 1\}$, $\theta_k = \rho_k^{l_k}$, $k = 1, 2$,

$\rho_1 = \rho_2 = e^{-\pi j}$ et $\bar{\rho} = \begin{pmatrix} 1 \\ \rho_1^{l_1} \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \rho_2^{l_2} \end{pmatrix}$. Ensuite, on obtient \bar{Z}_ρ grâce à (4.39):

$$\bar{Z}_\rho = \begin{bmatrix} 1 & 7 & -1 & 7 & 1 & -7 & -1 & -7 \\ 0 & 4 & -2 & 2 & 2 & 2 & 0 & 4 \end{bmatrix}.$$

La DFT inverse mène à $\bar{Z}_a = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 7 & 0 & 0 \\ 0 & 3 & 1 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}$ qui finalement signifie :

$$adj(Z(\theta)) = \bar{Z}_a (\bar{\theta} \otimes I_2) = \begin{bmatrix} \theta_2 & 7\theta_1 \\ \theta_2 - \theta_1 & \theta_1\theta_2 + 3 \end{bmatrix}$$

4.2.4 L'utilisation de la fonction signe matricielle

En reprenant le choix du cas pair $\alpha = 2d$ dans la partie 2.2.2.4, c'est-à-dire en utilisant l'expansion fractionnelle partielle de (3.29) nous rappelons les itérations obtenues :

$$\begin{cases} Z_{k+1} = \frac{1}{\alpha} Z_k \sum_{m=1}^{\alpha} \left(\sin^2 \left(\frac{(2m+1)\pi}{4\alpha} \right) I_n + \cos^2 \left(\frac{(2m+1)\pi}{4\alpha} \right) Z_k^2 \right)^{-1} \\ Z_0 = Z \end{cases} \quad (4.43)$$

Elles peuvent être réécrites de la manière suivante :

$$\begin{cases} Z_{k+1} = \frac{1}{\alpha} Z_k \cdot \sum_{m=1}^{\alpha} \frac{1}{\cos^2 \left(\frac{(2m+1)\pi}{4\alpha} \right)} \left(\tan^2 \left(\frac{(2m+1)\pi}{4\alpha} \right) I_n + Z_k^2 \right)^{-1} \\ Z_0 = Z \end{cases} \quad (4.44)$$

On rappelle également que $\lim_{k \rightarrow \infty} Z_k = \text{sign}(Z)$ et que la convergence de $\text{sign}(Z)$ peut être également garantie pour une valeur suffisamment grande de α .

Se basant sur les résultats précédemment montrés, nous présentons ici le résultat nous permettant de calculer la fonction signe matricielle au cas paramétrique.

Théorème 4.5 : Soit $Z(\theta) \triangleq \bar{Z} \times (\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n}$ une matrice paramétrique de la forme (4.24) alors :

$$\text{sign}(Z(\theta)) = \lim_{\alpha \rightarrow \infty} Z(\theta) \cdot \left(\frac{1}{\alpha} \bar{N}_{adj} \times \left(\left(\sum_{m=1}^{\alpha} \Theta_2(m, \theta) \cdot \bar{\theta} \right) \otimes I_n \right) \right) \quad (4.45)$$

En définissant :

$$N_m(\theta) = \tan^2 \left(\frac{(2m+1)\pi}{4\alpha} \right) I_n + Z(\theta)^2 \quad (4.46)$$

\bar{N}_{adj} représente la partie indépendante des paramètres m et θ de $adj(N_m(\theta))$ et :

$$\Theta_2(m, \theta) = \frac{1}{\cos\left(\frac{(2m+1)\pi}{4\alpha}\right)^2 \det(N_m(\theta))} \quad (4.47)$$

Preuve : La transformation discrète de Fourier est utilisée pour calculer l'inverse de $N(\theta, m)$. Nous utilisons également les itérations définies dans [KL95] et données par (4.44). Rappelons, voir 2.2.2.4, que 2 paramètres sont utiles pour approximer la fonction signe : k , le nombre d'itérations de l'algorithme (4.44) et α le nombre de termes de la somme utilisée à l'itération k . Augmenter k revient à chaque itération à devoir recalculer des inverses de matrice dépendant de l'itération précédente $Z_k(\theta)$, ce qui n'est pas souhaitable. On fixe alors $k=1$, ce qui fixe $Z_0(\theta)$ et ne fait dépendre les inverses que de m et de $Z_0(\theta)$; dans ce cas il est alors nécessaire de choisir une valeur suffisamment grande de n_d . Le signe s'écrit alors :

$$\text{sign}(Z(\theta)) = \frac{1}{\alpha} Z(\theta) \left(\sum_{m=1}^{\alpha} \frac{1}{\cos\left(\frac{(2m+1)\pi}{4\alpha}\right)^2} N_m(\theta)^{-1} \right) \quad (4.48)$$

avec

$$N_m(\theta) = \Theta_1(m) I_n + Z_0(\theta)^2, \quad \Theta_1(m) = \tan\left(\frac{(2m+1)\pi}{4\alpha}\right)^2 \quad (4.49)$$

$\Theta_1(m)$ peut être considéré comme un paramètre supplémentaire et intégré dans un vecteur de paramètres étendu $\bar{\delta}(\theta, m)$. L'inverse de la matrice $N_m(\theta)$ est calculé via la transformation de Fourier :

$$N_m(\theta)^{-1} = \frac{1}{\det(N_m(\theta))} \text{adj}(N_m(\theta)) = \frac{1}{\det(N_m(\theta))} \bar{N}_{adj} \times (\bar{\delta}(\theta, m) \otimes I_n) \quad (4.50)$$

Avec $\bar{\delta}(\theta, m) = \begin{pmatrix} 1 \\ \theta \\ \vdots \\ \theta^k \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \Theta_1(m) \\ \vdots \\ \Theta_1(m)^{k_2} \end{pmatrix}$ et suivant (3.29) $\bar{N}_{adj} = \frac{1}{\bar{m}} \bar{N}_\rho$ avec $\bar{m} = \prod_{k=1}^p m_k$ et

$m_k - 1$ les plus grandes puissances possibles des paramètres de $\bar{\delta}(\theta, m)$.

En posant $\Theta_2(m, \theta) = \frac{1}{\cos\left(\frac{(2m+1)\pi}{4\alpha}\right) \det(N_m(\theta))}$, l'expression du signe est maintenant

donnée par :

$$\text{sign}(Z(\theta)) = Z(\theta) \cdot \left(\frac{1}{\alpha} \left(\sum_{m=1}^{\alpha} \Theta_2(m, \theta) \bar{N}_{adj} \times (\bar{\delta}(\theta, m) \otimes I_n) \right) \right) \quad (4.51)$$

\bar{N}_{adj} et I_n sont évidemment indépendants des paramètres m et θ . (4.51) s'écrit également :

$$\text{sign}(Z(\theta)) = Z(\theta) \cdot \frac{1}{\alpha} \bar{N}_{adj} \left(\left(\sum_{m=1}^{\alpha} \Theta_2(m, \theta) \cdot \bar{\delta}(\theta, m) \right) \otimes I_n \right) \quad (4.52)$$

La somme dans (4.52) est donc aisément calculable sur le vecteur des paramètres θ en faisant varier simplement l'indice de la somme m . ■

Exemple 4.9 : Calculons le signe matriciel de la matrice paramétrique suivante :

$$Z(\theta) = \begin{bmatrix} \theta_1^2 + 1 & -2\theta_2 \\ \theta_1\theta_2 & 4 \end{bmatrix}$$

A partir de (4.49) et avec $\Theta_1(m) = \tan\left(\frac{(2m+1)\pi}{4\alpha}\right)$:

$$N_m(\theta) = \Theta_1(m)I_n + Z_0(\theta)^2 = \begin{bmatrix} \theta_1^4 - 2\theta_1\theta_2^2 + 2\theta_1^2 + \Theta_1(m) + 1 & -2\theta_1^2\theta_2 - 10\theta_2 \\ \theta_1^3\theta_2 + 5\theta_1\theta_2 & -2\theta_1\theta_2^2 + \Theta_1(m) + 16 \end{bmatrix}$$

$N_m(\theta)$ admet une décomposition minimale de la forme :

$$\bar{N} = [N_1 \quad N_2 \quad \cdots \quad N_9] = \begin{bmatrix} 1 & 0 & 0 & -10 & 1 & 0 & 0 & 0 & -2 & 0 & 2 & 0 & 0 & -2 & 0 & 0 & 1 & 0 \\ 0 & 16 & 0 & 0 & 0 & 1 & 5 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

et $\bar{\delta}(\theta, m) = [1 \quad \theta_2 \quad \Theta_1(m) \quad \theta_1\theta_2 \quad \theta_1\theta_2^2 \quad \theta_1^2 \quad \theta_1^2\theta_2 \quad \theta_1^3\theta_2 \quad \theta_1^4]^T$.

Ensuite, en appliquant la méthode DFT développée dans (4.39), \bar{N}_{adj} s'écrit :

$$\bar{N}_{adj} = \begin{bmatrix} N_{adj_1} & \cdots & N_{adj_9} \end{bmatrix} = \begin{bmatrix} 16 & 0 & 0 & 10 & 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & -5 & 0 & 0 & -2 & 0 & 2 & 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}$$

Puis, le déterminant est donné par :

$$\det(N_m(\theta)) = \theta_1^4 \Theta_1(m) + 16\theta_1^4 + 16\theta_1^3 \theta_2^2 + 2\theta_1^2 \Theta_1(m) + 4\theta_1^2 \theta_2^4 + 32\theta_1^2 - 4\theta_2^3 \Theta_1(m) + 16\theta_1 \theta_2^2 + \Theta_1(m)^2 + 17\Theta_1(m) + 16$$

Le nombre de termes de la somme choisi est $\alpha = 18$. A l'aide de (4.45), le signe de $Z(\theta)$ est donné par :

$$\begin{aligned} \text{sign}(Z(\theta)) &= Z(\theta) \cdot \left(\frac{1}{\alpha} \bar{N}_{adj} \times \left(\left(\sum_{m=1}^{\alpha} \Theta_2(m, \theta) \cdot \bar{\theta} \right) \otimes I_n \right) \right) \\ \text{sign}(Z(\theta)) &\approx \frac{1}{18} \begin{bmatrix} \theta_1^2 + 1 & -2\theta_2 \\ \theta_1 \theta_2 & 4 \end{bmatrix} \cdot \left(\bar{N}_{adj} \times \left(\left(\sum_{m=1}^{18} \Theta_2(m, \theta) \cdot \bar{\theta} \right) \otimes I_2 \right) \right) \end{aligned} \quad (4.53)$$

où $\bar{\theta} = \bar{\delta}(\theta, m) = [1 \quad \theta_2 \quad \Theta_1(m) \quad \theta_1 \theta_2 \quad \theta_1 \theta_2^2 \quad \theta_1^2 \quad \theta_1^2 \theta_2 \quad \theta_1^3 \theta_2 \quad \theta_1^4]^T$ et $\Theta_2(m, \theta)$ est donné par (4.47).

Pour montrer la pertinence de la méthode proposée, la figure suivante (Fig. 4.6) représente l'erreur en norme entre la matrice signe paramétrique (4.53) évaluée sur un ensemble de points fixé du paramètre θ , et les fonctions signes matricielles calculées à l'aide des itérations de Newton pour chaque paramètre θ fixé, en fonction des paramètres θ_1 et θ_2 .

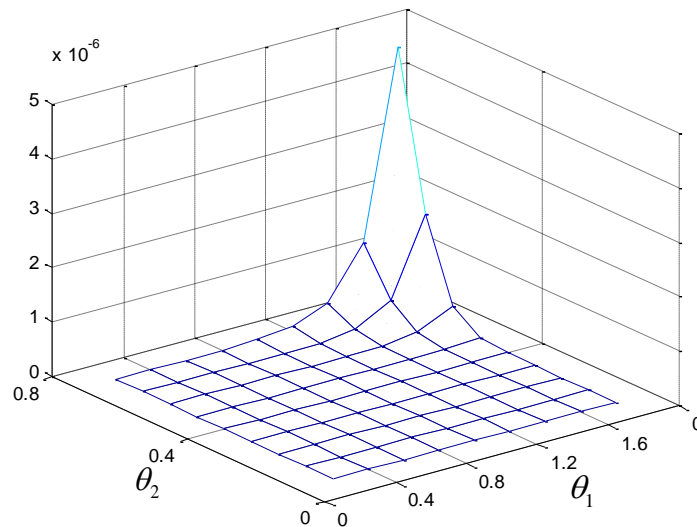


Fig. 4.6 : L'erreur en fonction des paramètres θ_1 , θ_2 .

Deux applications pour des systèmes d'ordre supérieurs sont présentées dans les deux sections suivantes, appliquées à la démarche développée dans cette partie.

4.2.5 Un filtre de Kalman paramétrique estimant les harmoniques d'amplitude d'ondes.

Le but de cet exemple est d'estimer les harmoniques d'un signal périodique de période $1/50\text{Hz}$. Ce problème considère un unique paramètre w . Dans le cas général, le système est donné par :

$$\begin{cases} A(w) = \text{diag}(J_1, J_2, \dots, J_p), B = \begin{bmatrix} 0_{(2p-2,2)} \\ I_2 \end{bmatrix} \\ C = \begin{bmatrix} 1 & 0 & 1 & 0 & \dots & 1 & 0 \end{bmatrix} \end{cases}$$

où diag signifie une matrice bloc-diagonale et $0_{(2p-2,2)} \in \mathbb{R}^{(2p-2) \times 2}$ une matrice avec entrées nulles. $p-1$ est le nombre d'harmoniques considéré dans ce problème. Les matrices d'harmonique J_i , $i \in \{1, \dots, p\}$ sont données par : $J_i = \begin{bmatrix} 0 & \sigma_i w \\ -\sigma_i w & 0 \end{bmatrix}$ où σ_i est un nombre constant choisi de manière arbitraire pour chaque harmonique.

Considérons maintenant le cas de deux harmoniques avec $\sigma_1 = 1$, $\sigma_2 = 3$, $\sigma_3 = 5$. Le problème consiste en la résolution de l'équation de Riccati paramétrique suivante :

$$A(w)X(w) + X(w)A(w)^T - X(w)C^T C X(w) + 20I_6 = 0 \quad (4.54)$$

Dans ce cas, la matrice Hamiltonienne correspondante s'écrit :

$$H(w) = \begin{bmatrix} \text{diag}(J_1, J_2, J_3)^T & -C^T C \\ -20I_6 & -\text{diag}(J_1, J_2, J_3) \end{bmatrix}$$

A partir de (4.49), la matrice $N_m(w) = \begin{bmatrix} N_{11}(w, m) & N_{12}(w, m) \\ N_{21}(w, m) & N_{22}(w, m) \end{bmatrix}$ s'écrit :

$$N_{11}(w, m) = N_{22}(w, m) =$$

$$\begin{bmatrix} -w^2 + \Theta_1(m) + 20 & 0 & 20 & 0 & 20 & 0 \\ 0 & \Theta_1(m) - w^2 & 0 & 0 & 0 & 0 \\ 20 & 0 & -9w^2 + \Theta_1(m) + 20 & 0 & 20 & 0 \\ 0 & 0 & 0 & \Theta_1(m) - 9w^2 & 0 & 0 \\ 20 & 0 & 20 & 0 & \beta & 0 \\ 0 & 0 & 0 & 0 & 0 & \Theta_1(m) - 25w^2 \end{bmatrix}$$

$$\beta = -25w^2 + \Theta_1(m) + 20,$$

$$N_{12}(w, m) = \begin{bmatrix} 0 & w & 0 & 3w & 0 & 5w \\ -w & 0 & -w & 0 & -w & 0 \\ 0 & w & 0 & 3w & 0 & 5w \\ -3w & 0 & -3w & 0 & -3w & 0 \\ 0 & w & 0 & 3w & 0 & 5w \\ -5w & 0 & -5w & 0 & -5w & 0 \end{bmatrix},$$

$$N_{21}(w, m) = \begin{bmatrix} 0 & 40w & 0 & 0 & 0 & 0 \\ -40w & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 120w & 0 & 0 \\ 0 & 0 & -120w & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 200w \\ 0 & 0 & 0 & 0 & -200w & 0 \end{bmatrix},$$

et $\Theta_1(m) = \tan\left(\frac{(2m+1)\pi}{4\alpha}\right)^2$. $N_m(w)$ admet une décomposition minimale de la forme :

$$\bar{N} = [N_1 \ N_2 \ N_3 \ N_4] \text{ et } \bar{\delta}(w, m) = [1 \ \Theta_1(m) \ w \ w^2]^T.$$

Ensuite, en appliquant la méthode DFT, \bar{N}_{adj} est déterminée et admet une forme minimale donnée par : $\bar{N}_{adj} = [N_{adj_1} \ \cdots \ N_{adj_{20}}]$ et le vecteur associé : $\bar{\theta} = \bar{\delta}(w, m) \in \mathbb{R}^{20}$. Pour le calcul du signe, nous avons choisi la méthode donnée par (4.45) avec $\alpha = 66$. Le signe de $H(w)$ est donné par :

$$\text{sign}(H(w)) = H(w) \cdot \left(\frac{1}{66} \bar{N}_{adj} \times \left(\sum_{m=1}^{66} \Theta_2(m, \theta) \cdot \bar{\theta} \right) \otimes I_{12} \right) \quad (4.55)$$

où $\Theta_2(m, w)$ est donné par (4.47). Les figures suivantes résument les résultats obtenus. Pour montrer la précision de la matrice solution $X(w)$ extraite de la matrice $\text{sign}(H(w))$ dans (4.55), la Fig. 4.7 donne l'erreur en norme :

$$er = \left\| A(w)X(w) + X(w)A(w)^T - X(w)C^T CX(w) + 20.I_6 \right\|_2$$

évaluée sur un ensemble de paramètres fixés de w . La figure montre clairement que $er \leq 7e^{-6}$ pour tout $w \in [0.1, 1]$.

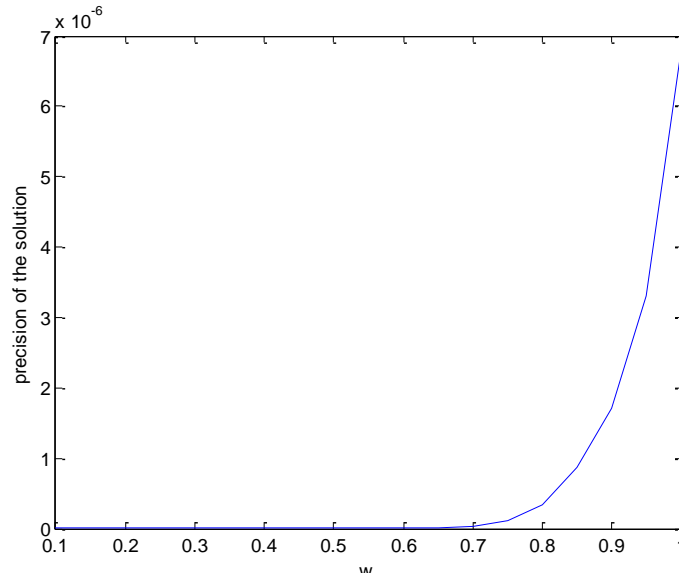


Fig. 4.7 : L'erreur en fonction du paramètre w .

De plus, la Fig. 4.8 montre l'évolution des parties réelles des valeurs propres de $X(w)$. Nous pouvons observer que $X(w)$ est définie positive pour tout $w \in [0.1, 1]$.

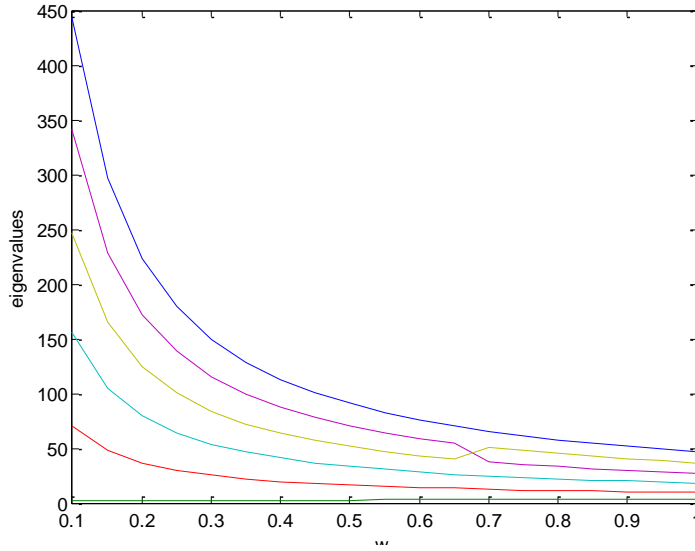


Fig. 4.8 : Evolution de la partie réelle des valeurs propres de la solution $X(w)$.

4.2.6 Modèle paramétré d'une aile flottante

Le modèle considéré dans cette partie a été introduit par [vWin08] dans la section 4.9. Il s'agit d'un modèle quasi-statique 2-D tiré d'une aile « intelligente », où des actionneurs sont sur les volets de l'aile. Le modèle a quatre états, un de contrôle en entrée (angle du volet), et deux mesures de sortie (angle d'attaque et déplacement vertical) et est dépendant de manière polynomiale selon la vitesse du vent v . Le modèle se présente sous la forme suivante :

$$\begin{cases} \dot{x} = (A_1 + A_2 v + A_3 v^2) x + B_3 v^2 u \\ y = C_1 x \end{cases}$$

avec l'état $x = [h \quad \alpha \quad \dot{h} \quad \dot{\alpha}]^T$. Les matrices sont données par :

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{I_\alpha}{q_1} k_h & \frac{-m_w x_\alpha b}{q_1} k_\alpha & \frac{I_\alpha}{q_1} c_h & \frac{-m_w x_\alpha b}{q_1} c_\alpha \\ \frac{-m_w x_\alpha b}{q_1} k_h & \frac{m_l}{q_1} k_\alpha & \frac{-m_w x_\alpha b}{q_1} c_h & \frac{m_l}{q_1} c_\alpha \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{I_\alpha}{q_1} q_2 + \frac{m_w x_\alpha b}{q_1} q_4 & \frac{I_\alpha}{q_1} q_2 q_6 + \frac{m_w x_\alpha b}{q_1} q_4 q_6 \\ 0 & 0 & \frac{-m_w x_\alpha b}{q_1} q_2 + \frac{-m_t}{q_1} q_4 & \frac{-m_w x_\alpha b}{q_1} q_2 q_6 + \frac{-m_t}{q_1} q_4 q_6 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{I_\alpha}{q_1} q_2 + \frac{m_w x_\alpha b}{q_1} q_4 & 0 & 0 \\ 0 & \frac{-m_w x_\alpha b}{q_1} q_2 + \frac{-m_t}{q_1} q_4 & 0 & 0 \end{bmatrix},$$

$$B_3 = \begin{bmatrix} 0 \\ 0 \\ \frac{I_\alpha}{q_1} q_3 + \frac{m_w x_\alpha b}{q_1} q_5 \\ \frac{-m_w x_\alpha b}{q_1} q_3 + \frac{-m_t}{q_1} q_5 \end{bmatrix},$$

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Pour la valeur des paramètres, voir sections 4.9.1 et 4.9.2 dans la table 4.4 [vWin08]. On pourra y trouver également une représentation schématique du modèle de l'aile « intelligente » correspondante.

Le but ici est de trouver un contrôleur $u = -K(\nu)x$ qui stabilise le système pour toutes les valeurs de $\nu \in [5, 15] m.s^{-1}$. L'idée est donc de résoudre une équation de Riccati dépendante du paramètre ν à l'aide de la fonction signe matricielle. Ensuite, le contrôleur sera construit suivant l'expression : $u = -B^T(\nu)X(\nu)x$. La matrice Hamiltonienne correspondante à l'équation de Riccati s'écrit :

$$H = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -232.45 & 0.38-0.08v^2 & -2.24-0.08v & 0.004-0.01v & 0 & 0 & -0.0019v^4 & -0.0079v^4 \\ 381.88 & -51.52-0.33v^2 & 3.68-0.33 & -0.65-0.05v & 0 & 0 & -0.0079v^4 & -0.033v^4 \\ -1 & 0 & 0 & 0 & 0 & 0 & 232.45 & -381.88 \\ 0 & -1 & 0 & 0 & 0 & 0 & -0.38+0.08v^2 & 51.52+0.33v^2 \\ 0 & 0 & 0 & 0 & -1 & 0 & 2.24+0.08v & 3.68-0.33v \\ 0 & 0 & 0 & 0 & 0 & -1 & -0.005+0.01v & 0.65+0.05v \end{bmatrix}$$

En suivant les mêmes démarches que pour l'exemple précédent avec $\alpha=16$, la solution $X(v)$ vérifiant l'équation de Riccati a une erreur en norme :

$$er = \|A(v)X(v) + X(v)A(v)^T - X(v)B(v)B^T(v)X(v) + C^T C\|_2$$

de l'ordre de 10^{-7} . La figure suivante montre la comparaison entre les valeurs propres en boucle ouverte et boucle fermée pour des valeurs de v comprises entre 5 et 15 $m.s^{-1}$.

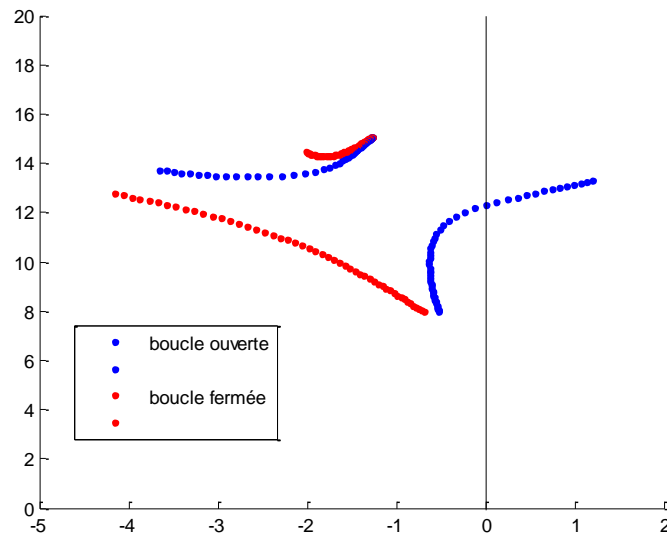


Fig. 4.9 : Evolution des valeurs propres en boucle ouverte et fermée.

On observe bien sur cette figure (Fig. 4.9) que toutes les valeurs propres en boucle fermée pour les valeurs de v comprises entre 5 et 15 sont bien dans le plan complexe négatif, le correcteur $u = -K(v)x$ calculé assure bien la stabilité du système.

La dernière partie de ce chapitre se consacre à l'étude du domaine de stabilité d'une matrice paramétrique en utilisant la fonction signe matricielle étendue. La particularité de l'application nécessitant des calculs sur des frontières de stabilité fait que la notion de fonction signe matricielle étendue sera utilisée.

4.3 La fonction signe matricielle étendue pour l'étude du domaine de stabilité

4.3.1 Introduction

La méthode présentée repose sur la généralisation de la définition de la fonction signe matricielle, décrite dans le chapitre 2. Cette extension a pour but de pouvoir traiter des problèmes d'étude du domaine de stabilité d'une matrice à dépendance paramétrique. Cela passera par une étude de la stabilité robuste et également par la résolution d'équations de Lyapunov.

4.3.2 La fonction signe étendue

La fonction signe matricielle n'est pas définie pour des matrices contenant des valeurs propres nulles ou imaginaires pures. De plus, une méthode itérative de calcul de la forme :

$$\begin{cases} Z_0 = Z \\ Z_{k+1} = f_R(Z_k) \\ \text{sign}(Z) = Z_k, \quad k \rightarrow \infty \end{cases} \quad (4.56)$$

où f_R est une fonction rationnelle préservant le signe (c.-à-d. $\text{sign } f_R(Z) = \text{sign}(Z)$) et globalement convergente peut devenir non-convergente (ou avoir une convergence très lente) quand Z a des valeurs propres proches de l'axe des imaginaires (voir [Mat78]). En fait, considérons par exemple les itérations de Newton, pour le cas où Z a une valeur propre proche de l'axe des imaginaires $\sigma_0 \approx j\omega$, et l'algorithme itératif de Newton induit :

$$\sigma_{k+1} = \frac{1}{2} \left(\sigma_k + \frac{1}{\sigma_k} \right), \quad \sigma_0 \approx j\omega \quad (4.57)$$

où σ_k est la valeur propre correspondante de Z_k . Donc, σ_k est purement imaginaire pour tout k . En utilisant $\sigma_k = jv_k$, $v_k \in \mathbb{R}$ on a :

$$v_{k+1} = \frac{1}{2} \left(v_k - \frac{1}{v_k} \right), \quad v_0 \approx \omega \quad (4.58)$$

Ces itérations deviennent donc non-convergentes. Une voie pour éviter cette difficulté est d'utiliser ce que l'on appelle la fonction signe matricielle étendue, introduite dans [Mat78]. Dans cette étude, Mattheys l'utilise pour séparer les modes oscillants stables, instables et non amortis d'un système linéaire à temps invariant (LTI).

La fonction signe étendue notée (*exsign*) dans la suite, est définie pour $z \in \mathbb{C}$ par :

$$\text{exsign}(z) = \begin{cases} 1, & \text{if } \text{Re}(z) > 0 \\ 0, & \text{if } \text{Re}(z) = 0 \\ -1, & \text{if } \text{Re}(z) < 0 \end{cases} \quad (4.59)$$

Rappelons les éléments cités dans le Chapitre 2 sur cette fonction, la fonction exsign est discontinue et il est impossible d'utiliser la décomposition de Jordan pour l'extension au cas matriciel. Dans le but de définir $\text{exsign}(Z)$, $Z \in \mathbb{R}^{n \times n}$, il faut considérer un scalaire positif suffisamment petit ε tel que $\text{sign}(Z \pm \varepsilon I_n)$ soient tous les deux définis. $\text{exsign}(Z)$ est ensuite donné par :

$$\text{exsign}(Z) = \frac{1}{2} (\text{sign}(Z + \varepsilon I) + \text{sign}(Z - \varepsilon I)) \quad (4.60)$$

Un choix particulier de ε est donné dans [SP06] :

$$\varepsilon < \min_i |\lambda_i^*| \quad (4.61)$$

avec $\lambda_i^* \neq 0$ correspondant aux valeurs propres non nulles de Z . Concrètement, ce choix a l'effet suivant : les valeurs propres $\lambda_i \in \Lambda(Z)$ avec leur partie réelle dans $[-\varepsilon, \varepsilon]$ sont mises à zéro dans $\text{exsign}(Z)$. Un choix possible de ε est basé sur l'inverse de Drazin de Z . En fait, comme $\|Z^D\|_F \geq \frac{1}{\min_i |\lambda_i^*|}$, on peut utiliser la valeur suivante :

$$\varepsilon = \frac{\mu}{\|Z^D\|_F}, \quad (4.62)$$

où $0 < \mu < 1$ et Z^D est l'inverse de Drazin de Z qui est défini comme l'unique solution des équations $XZ = ZX$, $XZX = X$ et $Z^{q+1}X = Z^q$ où q est l'indice de Z (c.-à-d. le plus petit entier positif tel que $\text{rang}(Z^{q+1}) = \text{rang}(Z^q)$).

Remarque 4.9 : Il existe une multitude d'algorithmes itératifs pour le calcul de l'inverse de Drazin. Considérant la spécificité du problème que nous traitons et la nécessité d'étendre nos résultats au cas dépendant des paramètres, nous privilégierons ceux qui n'utilisent pas de calcul de valeurs propres.

Exemple 4.10 : Considérons la matrice suivante :

$$Z = \begin{bmatrix} -3 & 1 & 1 & 1 \\ -4 & -1 & 2.4 & 0.8 \\ -6 & 0 & 2.4 & 1.8 \\ -2 & 0 & 0.8 & 0.6 \end{bmatrix}$$

avec le spectre $\Lambda(Z) = \{\pm\sqrt{3}i, -1, 0\}$. Prenant $\mu = 0.99$, le calcul de ε (4.62) mène à $\varepsilon = 0.2113$. $\text{sign}(Z \pm \varepsilon I_n)$ sont tous les deux trouvés en utilisant l'algorithme de Newton. Seulement 9 itérations sont nécessaires pour chaque fonction signe pour obtenir le résultat suivant :

$$\text{exsign}(Z) = \begin{bmatrix} 0 & 1 & -0.8 & 0.4 \\ 0 & 0 & 0 & 0 \\ 0 & 1.5 & -1.2 & 0.6 \\ 0 & 0.5 & -0.4 & 0.2 \end{bmatrix}$$

alors que $\text{sign}(Z)$ n'est pas défini.

Comme indication, notons que $\Lambda(\text{exsign}(Z)) = \{0, 0, 0, -1\}$, ce qui veut dire qu'une valeur propre de Z appartient à \mathbb{C}_- , et les trois autres sont sur l'axe imaginaire.

4.3.3 La fonction signe matricielle paramétrique

Nous rappelons ici que l'objectif est de trouver une bonne approximation de la fonction signe matricielle de matrices paramétrées. Le but est donc de trouver le signe étendu $\text{exsign}(Z(\theta))$. Nous allons donc généraliser la partie précédente au cas paramétrique.

4.3.3.1 Inverse de Drazin généralisée

Quelques travaux dans le but de généraliser l'inverse de Drazin au cas paramétrique peuvent être trouvés par exemple dans [BW04] et les références citées dans cet article. Dans la suite, nous proposons une généralisation, au cas paramétrique affine, de l'algorithme 2 dans [BW04]. Le choix de cet algorithme est basé sur le fait qu'il ne requiert aucun calcul de valeurs propres. Redéfinissons la forme classique souvent utilisée soit dans sa forme rationnelle ou polynomiale dans ce mémoire de thèse :

$$Z(\theta) \triangleq \bar{Z}(\bar{\theta} \otimes I_n) \in \mathbb{R}^{n \times n} \quad (4.63)$$

L'algorithme proposé est le suivant :

Algorithme 4.3 :

- Etape 1 : Calculer la séquence de polynômes $\{z_1(\theta), \dots, z_n(\theta)\}$ et la séquence de matrices $\{B_1(\theta), \dots, B_{n-1}(\theta)\}$ telles que :

$$\begin{aligned} z_1(\theta) &= -\text{trace}(Z(\theta)) \\ B_1(\theta) &= Z(\theta) + z_1(\theta)I_n \end{aligned}$$

et

$$\begin{aligned} z_j(\theta) &= -\frac{1}{j} \left(\text{trace}(Z(\theta)B_{j-1}(\theta)) \right), \quad j = 2, 3, \dots, n \\ B_j(\theta) &= Z(\theta)B_{j-1}(\theta) + z_j(\theta)I_n, \quad j = 2, 3, \dots, n-1 \end{aligned}$$

- Etape 2 : Trouver le plus grand entier t et le plus petit entier r qui satisfont les deux équations suivantes :

$$\begin{cases} z_t(\theta) \neq 0 \\ B_r(\theta) = 0_n \end{cases}$$

- Etape 3 : Poser $k = r - t$, et calculer $Z^D(\theta)$ comme suit :

$$Z^D(\theta) = \frac{(-1)^{k+1}}{z_t(\theta)^{k+1}} Z(\theta) B_{t-1}(\theta)^{k+1} \quad (4.64)$$

Remarque 4.10 : Notons que les étapes 1 et 3 de l'Algorithme 4.3 peuvent être réalisées utilisant l'arithmétique associée à la classe de matrice paramétrée considérée et décrite dans les chapitres précédents.

4.3.3.2 Fonction signe matricielle paramétrique étendue

Nous proposons ici une généralisation de la fonction signe matricielle étendue au cas paramétrique. On définit $\text{exsign}(Z(\theta))$ par :

$$\text{exsign}(Z(\theta)) = \frac{1}{2} \left(\text{sign}(Z(\theta) + \varepsilon(\theta)I_n) + \text{sign}(Z(\theta) - \varepsilon(\theta)I_n) \right) \quad (4.65)$$

avec :

$$\varepsilon(\theta) = \frac{\mu}{\|Z^D(\theta)\|_F}, \quad 0 < \mu < 1 \quad (4.66)$$

où $Z^D(\theta)$ est calculée à l'aide de l'Algorithme 4.3.

4.3.4 Un algorithme pour la matrice signe paramétrique

La précédente introduction de quelques outils comme l'inverse de Drazin d'une matrice paramétrique ou la fonction signe étendue nous permet de présenter l'algorithme principal de cette section.

Algorithme 4.4 :

- Etape 1 : Initialisation

Définir un ensemble de bornes $\theta_k \in [\underline{\theta}_k, \bar{\theta}_k]$, $k \in \{1, \dots, l\}$. Définir une séquence de nœuds $\{d_1^k, \dots, d_N^k\}$, $k \in \{1, \dots, l\}$ où $\underline{\theta}_k = d_1^k < \dots < d_N^k = \bar{\theta}_k$ et noter Δ_t , $t \in \{1, \dots, N^l\}$ les domaines définis par l'ensemble d'intervalles $[d_i^k, d_{i+1}^k]$ où $i \in \{1, 2, \dots, N-1\}$ et $k \in \{1, \dots, l\}$.

- Etape 2 : Calculer $Z^D(\theta)$ en utilisant l'Algorithme 4.3. Pour déterminer le scalaire μ tel que $0 < \mu < 1$, calculer $\varepsilon(\theta)$ se basant sur (4.62). Poser $j=1$, et définir $\bar{Z}_0(\theta) = Z(\theta) + \varepsilon(\theta)I_n$ et $\underline{Z}_0(\theta) = Z(\theta) - \varepsilon(\theta)I_n$.
- Etape 3 : A l'itération j de l'algorithme, calculer $\bar{Z}_j = [\bar{P}_{hg}]_{1 \leq h \leq n, 1 \leq g \leq n}$ (respectivement $\underline{Z}_j = [\underline{P}_{hg}]_{1 \leq h \leq n, 1 \leq g \leq n}$) sur Δ_t , $t \in \{1, \dots, N^l\}$ tel que $\bar{Z}_j \in S_l^{1,c}$ (respectivement $\underline{Z}_j \in S_l^{1,c}$) où $S_l^{1,c}$ est défini de la manière suivante :

$$S_l^{1,c} = \left\{ \begin{array}{l} [\bar{P}_{hg}]_{1 \leq h \leq n, 1 \leq g \leq n} / \bar{P}_{hg} \in C^1(\mathbb{R}^l), \\ \bar{P}_{hg} \text{ est un polynôme de degré } c \text{ sur chaque } \Delta_t, t \in \{1, \dots, N^l\} \end{array} \right\}$$

- Etape 4 : Pour un scalaire suffisamment petit δ , évaluer le critère de convergence : $J = \|\bar{Z}_j - \bar{Z}_{j-1}\|_F + \|\underline{Z}_j - \underline{Z}_{j-1}\|_F$, si $J > \delta$ retourner à l'Etape 3, sinon calculer $exsign(Z(\theta)) = \frac{1}{2}(\bar{Z}_j + \underline{Z}_j)$.

Remarque 4.11 : Alors qu'un algorithme de Newton est proposé dans l'Algorithme 4.4, n'importe quelle méthode de la forme $Z_{t+1} = f_R(Z_t)$ avec f_R étant une fonction rationnelle préservant le signe et globalement convergente peut être utilisée. Cela peut avoir un effet significatif sur la vitesse de convergence de l'Algorithme 4.4.

Remarque 4.12 : La vitesse de convergence dépend évidemment du nombre de nœuds et du degré des polynômes utilisés. Les polynômes de Hermite semblent plus adéquats prenant en compte les discontinuités de la fonction signe, et pourraient donc être utilisés.

Testons d'abord sur des exemples basiques la capacité de l'Algorithme 4.4 à identifier des régions de stabilité convexes ou concaves de certaines matrices paramétrées conçues pour ce test.

Exemple 4.11 : Considérons les matrices paramétrées Z_1 et Z_2 :

$$Z_1(\theta) = \begin{bmatrix} -(0.9 - \theta_1)(1.9 - \theta_2) + (\theta_1 + \theta_2)^2 & -1 \\ 0 & -(0.9 - \theta_1)(0.9 - \theta_2) + \theta_1^2 \end{bmatrix}$$

$$Z_2(\theta) = \begin{bmatrix} (1.7 + \theta_1)(1.9 + \theta_2) & -1 \\ 0 & -(0.9 - \theta_1)(0.9 - \theta_2) + \theta_1^2 \end{bmatrix}$$

avec $\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$. La Fig. 4.10 (resp. Fig 4.11) présente dans le plan (θ_1, θ_2) le domaine de stabilité pour $Z_1(\theta)$ (resp. $Z_2(\theta)$). Elle correspond à l'intersection des 2 zones définies par les termes diagonaux de $Z_1(\theta)$ (resp. $Z_2(\theta)$).

Comme attendu, l'Algorithme 4.4 mène à une région de stabilité convexe pour Z_1 , en bleu dans la Fig. 4.10, et concave pour la région de stabilité de Z_2 , en bleu dans Fig. 4.10.

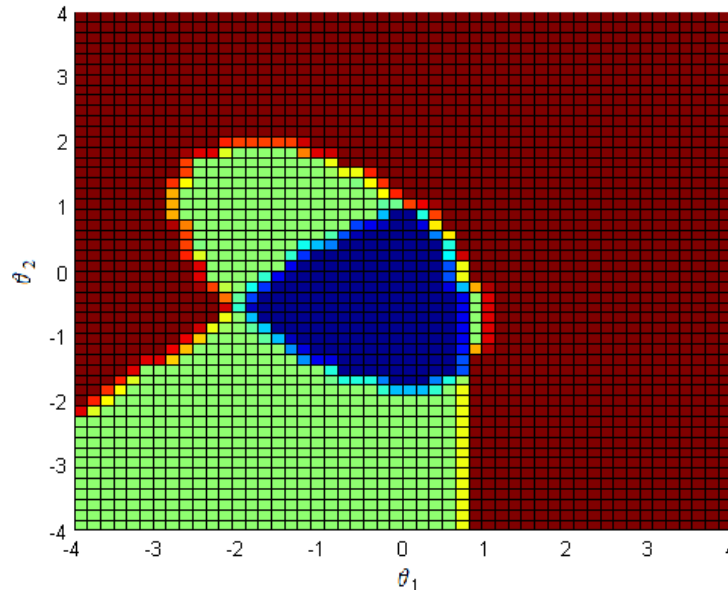


Fig. 4.10 : Région de Stabilité pour $Z_1(\theta)$

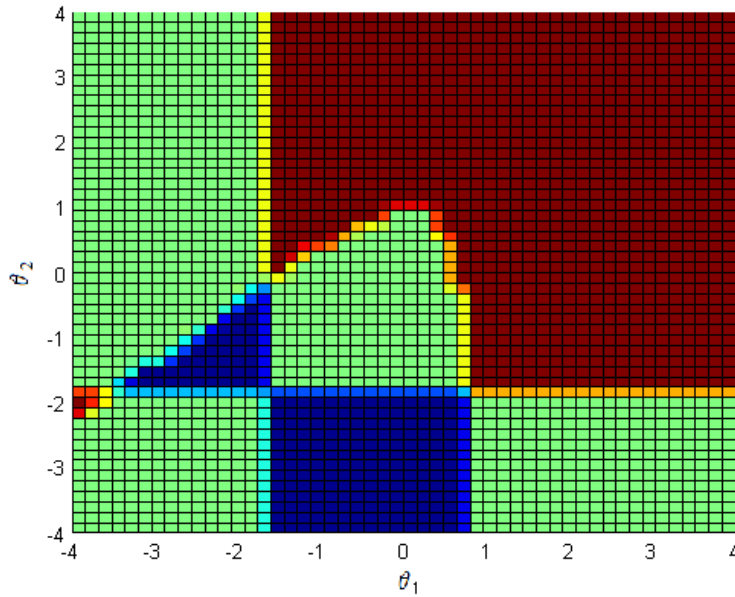


Fig. 4.11 : Région de Stabilité pour $Z_2(\theta)$.

En dehors des régions en bleu, $\text{exsign}(Z_1(\theta)) \neq -I$ (resp. $\text{exsign}(Z_2(\theta)) \neq -I$) et donc $Z_1(\theta)$ et $Z_2(\theta)$ sont instables.

L'Algorithme 4.4, proposé dans la section précédente, est testé sur des exemples repris de la base de données COMPI_{ib} [Lei04]. Il s'agit d'une librairie composée de différents modèles LTI et comprenant des problèmes purement académiques et également des exemples industriels plus réalistes. Il existe par exemple des modèles d'avion (AC), d'hélicoptère (HE), et des tests académiques (NN), *etc...*

Cette librairie est aujourd'hui utilisée principalement dans le but d'évaluer l'efficacité de certaines stratégies de contrôle (voir par exemple [YS11]).

Les Figures 4.12 et 4.13 montrent respectivement les domaines de stabilité trouvés par l'Algorithme 4.4 pour les exemples NN5 ($n=7, m=1$ et $p=2$) et AC7 ($n=9, m=1$ et $p=2$). La fonction signe matricielle étendue de $Z(\theta) = A + BKC$, $\theta = [k_1 \ k_2]^T = K^T$ a été obtenue pour ces exemples avec $c=3$, $\mu=0.9$ et $N=20$. La convergence est obtenue après 15 itérations de l'Algorithme 4.4.

Exemple 4.12 :

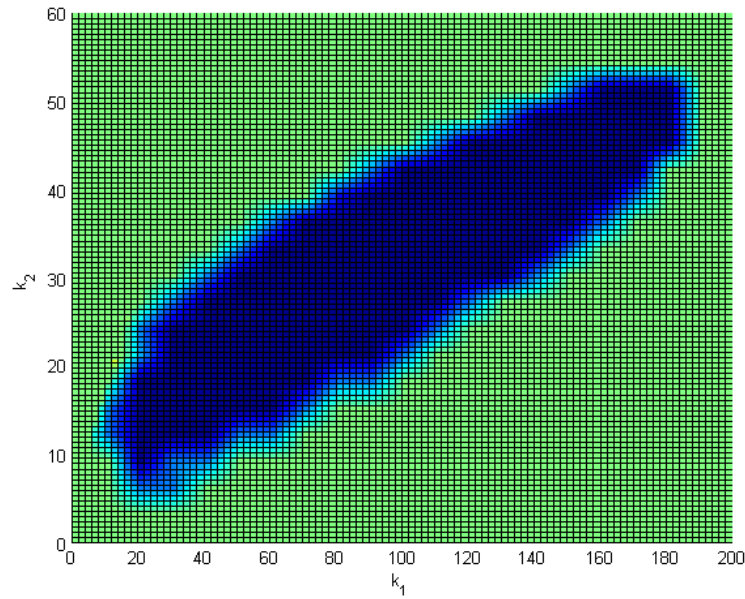


Fig. 4.12 : Région de stabilité (en bleu) utilisant l' Algorithme 4.4 pour NN5.

Exemple 4.13 :

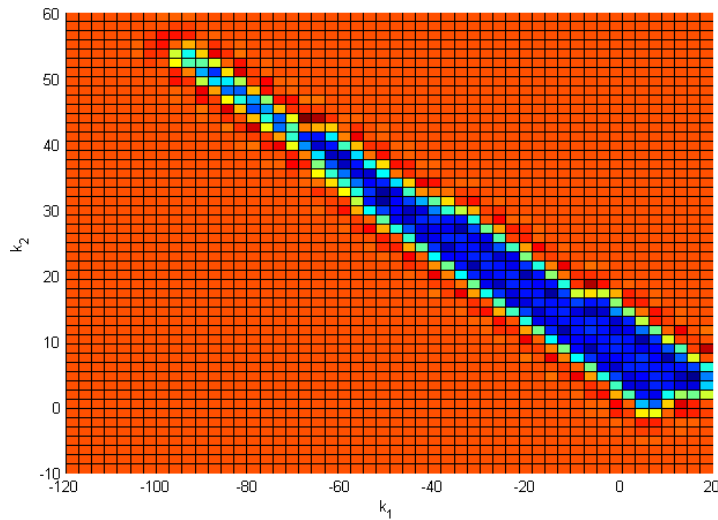


Fig. 4.13 : Région de stabilité (en bleu) utilisant l' Algorithme 4.4 pour AC7.

4.4 Conclusion

Cette méthode n'est pas comparable aux autres étant donné qu'elle sert ici à étudier des domaines de stabilité. L'utilisation de la fonction signe matricielle étendue nous a permis de traiter ce genre de problèmes, qui ne sont pas réalisables avec les fonctions signes normales. Cette méthode est la dernière étudiée dans ce manuscrit.

Chapitre 5 :

Conclusion

*Or, l'heure est au culte de l'intensité de l'instant,
à la valorisation de la satisfaction immédiate des désirs,
à la promotion de l'individualité.
La transmission des connaissances n'a, elle,
que lenteur, travail et exigence à proposer.*

Etienne Klein

Ce mémoire de thèse traite principalement d'une voie alternative au formalisme LMI pour l'analyse et la commande des systèmes paramétrés. L'objectif de ce travail a été de revoir la résolution des équations séminales sous-jacentes à ces problèmes d'analyse et de commande à l'aune de certaines méthodes utilisées jusqu'ici uniquement dans le cas non paramétrique. Le lien existant dans la littérature entre la notion de fonction signe matricielle et les équations de Riccati et de Lyapunov a inspiré l'idée d'étendre cette notion au cas dépendant des paramètres. Etant donné que cette notion a été intensivement utilisée dans ce travail de thèse, le premier chapitre du présent mémoire a consisté en un état de l'art sur la fonction signe matricielle, comprenant des définitions, propriétés et méthodes de calculs de cette fonction dans un premier temps puis sur le développement des liens existants entre la fonction signe matricielle et les problèmes considérés.

A partir de ce point, le manuscrit inclut trois chapitres sur des résolutions d'équations de Riccati, de Lyapunov et de Sylvester paramétrées. Le premier s'intéresse à des méthodes dites « non itératives ». Il faut entendre par cela que ces méthodes offrent un résultat direct. Le premier point a été dédié à la reformulation d'une équation de Lyapunov en une inversion matricielle. Le second point a pris en compte l'écriture intégrale de la fonction signe matricielle, celle-ci offrant un résultat « exact » du calcul de la fonction signe, menant à une solution elle-même « exacte » des équations de Lyapunov et de Riccati paramétrées. L'application de cette méthode aux problèmes d'analyse H_2 robuste et de commande H_2 structurée, des problèmes d'analyse et de commande d'un intérêt particulier, a été présentée. Quoique concluante l'extension de cette méthode à des cas paramétrés de dimensions élevées s'est avérée ardue. C'est pourquoi la dernière partie de ce chapitre a levé la nécessité de généraliser le calcul de la fonction signe matricielle en utilisant la définition intégrale. Il a été, néanmoins, possible de proposer une généralisation à l'ordre 2 de cette méthode de calcul de la fonction signe matricielle dans le cas dépendant des paramètres. Un verrou important consiste donc en l'extension de cette méthode de manière

générique au cas paramétré de plus grandes dimensions. La dernière partie de ce chapitre traite d'un autre problème ouvert dans l'automatique, la résolution d'équations de Sylvester paramétrées non standard, c'est-à-dire, ayant les matrices A et B de taille différentes. Une méthode a été proposée considérant deux cas distincts, la diagonalisation ou bloc-diagonalisation des matrices carrées A et B de l'équation de Sylvester. La démarche proposée est basée sur une combinaison de la fonction signe matricielle, pour trouver les valeurs et vecteurs propres d'une matrice, et sur une décomposition spectrale des matrices. Cette démarche originale a permis entre autre de mettre au clair un lien indirect entre la notion de fonction signe matricielle et les équations de Sylvester. Cette approche étant liée au calcul de la fonction signe matricielle par la définition intégrale, à l'instar de la méthode développée dans le chapitre précédent, elle souffre des mêmes inconvénients. C'est ainsi que ce travail de thèse s'est donc orienté vers la recherche de méthodes aboutissant à une solution approximée souvent à dépendance rationnelle en les paramètres.

Le chapitre 4 s'intéresse, pour essayer de contrer ce problème de dimension, à la résolution d'équations de Lyapunov et de Riccati paramétrées à l'aide de méthodes de calcul de la fonction signe matricielle itératives. Ces méthodes impliquent donc une approximation du calcul du signe, menant à une approximation rationnelle, et non irrationnelle comme on a pu le voir pour le cas de la définition intégrale appliquée à la résolution d'équations de Riccati. La première partie utilise l'expansion en polynômes de Laurent combinée avec les itérations de Newton-Shultz, ayant pour but principal d'éviter au maximum le calcul d'inversion de matrices paramétrées. Cela a permis de traiter des problèmes de plus grandes dimensions. Néanmoins, on observe une restriction sur la classe de matrices étudiées. Il est toujours possible de trouver un moyen permettant d'étendre cette classe de matrices (changement de variables, pondérations sur les matrices, etc.). Cependant, le souhait de développer une approche suffisamment générique et de complexité raisonnable a orienté la recherche vers d'autres démarches de calcul itératives. Ainsi, la seconde partie concerne une méthode de calcul de la fonction signe par des itérations de Padé particulières, combinée avec les inversions de Fourier (DFT). Cette méthode est la plus prometteuse, pour plusieurs raisons. Dans un premier temps, cette méthode réduit au maximum le nombre d'inversions, étant donné qu'elle se base sur des sommes pondérées incluant une seule inversion matricielle. Les itérations sont de plus ramenées à des itérations sur un vecteur, ce qui réduit considérablement la complexité de calcul. Cela nous a permis de traiter des problèmes d'ordre supérieurs à ceux étudiés précédemment. La dernière partie de ce chapitre traite d'une approche hybride symbolique/numérique. Cette autre voie possible de l'utilisation de la fonction signe matricielle, a été appliquée à la recherche de domaines de stabilité pour des systèmes paramétrés. Cette partie est basée sur la définition étendue de la fonction signe matricielle, où l'on a pris en compte le cas de valeurs propres à parties réelles nulles. Trouver un moyen pour affiner numériquement le calcul des inverses de Drazin inclus dans cette démarche serait une voie d'amélioration possible et aussi une question ouverte à traiter.

De futurs travaux dans la continuité de cette thèse pourraient dans un premier temps concerner une étude plus approfondie de la complexité des différentes méthodes développées. Dans un second temps, étant donné que cette thèse a une approche purement théorique de l'utilisation de la fonction signe matricielle, il serait intéressant d'appliquer certaines des méthodes développées les plus compétitives pour de plus grandes gammes de problèmes de commande. Par exemple, l'étude de la stabilité et/ou de la stabilisation de systèmes bidimensionnels, les résolutions d'équation de Riccati dépendantes de l'état (SDRE), ou encore au cas des régulateurs dits « re-réglables ». Un autre point important serait l'extension de la généralisation du calcul exact de la fonction signe matricielle pour des problèmes d'ordre supérieur. Finalement, une autre voie possible serait de faire le lien entre les sommes de carrés (SOS, de l'anglais « sum of square ») dans le cas matriciel et la fonction signe matricielle, au cas paramétrique ce qui permettrait d'ouvrir un grand champ d'applications aux méthodes basées sur le calcul de la fonction signe matricielle dans le cas dépendant des paramètres.

Bibliographie

*Je connais un critique qui est en même temps auteur...
ce qui le met en tant qu'auteur dans une situation critique !*

Raymond Devos

- [Aga90] P. Agathoklis. Algebraic necessary and sufficient conditions for the very strict Hurwitz property of a 2D Polynomial. *Multidimension. Systems and Signal Processing*, vol. 2, pp. 45-53, September 1990.
- [AG95] P. Apkarian, and P. Gahinet. A convex characterization of gain-scheduled H_∞ controllers. *IEEE Transactions on Automatic Control*, vol. 40, no. 5, pp. 853–863, 1995.
- [AM90] B. Anderson and J. B. Moore. Optimal control: linear quadratic methods, *Prentice-Hall*, Englewood Cliffs, N. J, 1990.
- [Bar94] B. R. Barmish. New tools for robustness of linear systems. *Macmillan Publishing Company*, 1994.
- [BD73] A.N. Beavers and E.D. Denman. A computational method for eigenvalues and eigenvectors of a matrix with real eigenvalues. *Numerical Mathematics*, vol. 21, pp. 389-396, 1973.
- [BD74] A.N. Beavers and E.D. Denman. Asymptotic solutions to the matrix Riccati equations. *Mathematical Biosciences*, vol. 20, pp. 339-344. 1974.
- [BD76] A.N. Beavers and E.D. Denman. A new method for the Lyapunov matrix equation. *Journal on Applied Mathematics, SIAM*, vol. 29, pp. 416-421, 1975.
- [BE-GF+94] S. Boyd, L. El Gaoi, E. Feron and V. Balakrishnan. Linear matrix inequalities in system and control theory. *Studies in Applied Mathematics*, vol. 15, 1994.
- [BHM97] R. Byers, C. He, and V. Mehrmann. The matrix sign function method and the computation of invariant subspaces. *Journal on Matrix Analysis and Applications, SIAM*, vol. 18, pp. 615–632, 1997.
- [BK95] B. Bakkaloglu and C. K. Koç. Parallel matrix iterations. *Proceedings of the 33rd Annual Allerton Conference on Communication, Control, and Computing*, pp. 440-446, 1995.
- [Bli04] P. A. Bliman. A convex approach to robust stability for linear systems with uncertain scalar parameters. *Journal of Control and Optimization, SIAM*, vol. 42, no. 6, pp. 2016–2042, 2004.

- [Bou08] A. Bouali. Approche descripteur pour l'analyse et la commande des systèmes LPV rationnels. *Thèse de doctorat*, université de Nantes, 278 pages, 2008.
- [BS72] R.H. Bartels and G.W. Stewart. Solution of the matrix equation $AX+XB=C$: Algorithm 432. *Communications of the ACM*, vol. 15, pp. 820–826, 1972.
- [BW04] F. Bu and Y. Wei. The algorithm for computing the Drazin inverses of two-variable polynomial matrices. *Applied Mathematics and Computation*, vol. 147, pp. 805–836, 2004.
- [CGT+05] G. Chesi, A. Garulli, A. Tesi, and A. Vicino. Polynomially parameter-dependent Lyapunov functions for robust stability of polytopic systems: an LMI approach. *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 365–370, 2005.
- [CGT+09] G. Chesi, A. Garulli, A. Tesi, and A. Vicino. Homogeneous polynomial forms for robustness analysis of uncertain systems. *Springer, Lecture Notes in Control and Information Sciences*, vol. 390, 2009.
- [Che08] G. Chesi. On the non-conservatism of a novel LMI relaxation for robust analysis of polytopic systems. *Automatica*, vol. 44, no. 11, pp. 2973–2976, 2008.
- [Che11] G. Chesi. Domain of attraction: analysis and control via SOS programming. *Lecture Notes in Control and Information Sciences*, vol. 415, 2011.
- [Çim08] T. Çimen. State-Dependent Riccati Equation (SDRE) control: a survey. *The International Federation of Automatic Control Seoul, Korea*, July 6–11, 2008.
- [DB76] E. D. Denman and A.N. Beavers. The matrix sign function and computations in systems. *Applied Mathematics and Computation*, vol. 2, pp. 63–94, 1976.
- [Den76] E. D. Denman. An additional algorithm for a system of coupled algebraic matrix Riccati equations. *IEEE Transaction Computation*, vol. C-25, pp. 91–93, 1976.
- [Din05] M. Dinh. Synthèse dépendant de paramètres par optimisation LMI de dimension finie : application à la synthèse de correcteurs reréglables. *Thèse de doctorat*, Université de Caen Basse-Normandie, 227 pages, 2005.
- [DSF+03] M. Dinh, G. Scorletti, V. Fromion, and E. Magarotto. Parameterized H_∞ controller design for adaptative trade-off finite dimensional LMI optimization. *European Control Conference*, Cambridge, UK, 2003.
- [DSF+05] M. Dinh, G. Scorletti, V. Fromion, and E. Magarotto. Parameter dependent H_∞ control by finite dimensional LMI optimization: Application to trade-off dependent control. *International Journal of Robust and Nonlinear Control*, vol. 15, no. 9, pp. 383–406, 2005.
- [DZG+94] J. Doyle, K. Zhou, K. Glover, and B. Bodenheimer. Mixed H_2 and H_∞ performance objectives II: *Optimal Control*, *IEEE Transactions on Automatic Control*, vol. 39, pp. 1575–1587, 1994.

- [EPA+05] Y. Ebihara, D. Peaucelle, D. Arzelier, and T. Hagiwara. Robust performance analysis of linear time-invariant uncertain systems by taking higher-order time-derivatives of the state, *Proceedings of Conference on Decision and Control*, pp. 5030-5035, 2005.
- [FAG96] E. Feron, P. Apkarian, and P. Gahinet. Analysis and synthesis of robust control systems via parameter-dependent Lyapunov functions. *IEEE Transactions on Automatic Control*, vol. 41, no. 7, pp. 1041–1046, 1996.
- [FB99] H. Fabbender and P. Benner. Initializing Newton’s method for discrete-time algebraic Riccati equations using the butterfly SZ algorithm. *IEEE International Symposium on Computer Aided Control System Design*, 1999.
- [Fer97] E. Feron. Analysis of robust H_2 performance using multiplier theory, *Journal on Control and Optimization, SIAM*, vol. 35, pp. 160-177, 1997.
- [FM76] E. Fornasini and G. Marchenisi. State-space realization theory of two-dimensional filters. *IEEE Transactions on Automatic Control*, vol. 21, no. 4, pp. 484-492, 1976.
- [Fri96] B. Friedland. Advanced control system design. *Prentice-Hall, Englewood Cliffs NJ*, pp. 110-112., 1996.
- [FYC11] Y. Feng, M. Yagoubi and P. Chevrel. Extended H_2 controller synthesis for continuous descriptor systems. *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1559-1564, 2011.
- [GAC96] P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent Lyapunov functions and real parametric uncertainty. *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 436–442, 1996.
- [Gar88] J. D. Gardiner. Iterative and parallel algorithms for the solution of algebraic Riccati equations. *Ph.D. dissertation, University of California ECE Department. Santa Barbara, CA*, 1988.
- [GL91] J. D. Gardiner and A. J. Laub. Parallel algorithms for algebraic Riccati equations. *International Journal of Control*, vol. 54, pp. 1317-1333, 1991.
- [GNV-L79] G. H. Golub, S. Nash, and C. F. Van Loan. A Hessenberg–Schur method for the problem $AX + XB = C$. *IEEE Transactions on Automatic Control*, vol. AC-4, pp. 909-913, 1979.
- [GPT+07] E. N. Goncalves, R. M. Palhares, R. H. C. Takahashi, and R. C. Mesquita. New strategy for robust stability analysis of discrete-time uncertain systems. *Systems and Control Letters*, vol. 56, no. 7, pp. 516–524, 2007.
- [GR65] J. Gradshteyn and I. Ryzhik. Table of integrals, series, and products. *Academic Press, New York*, 4th edition, 1965.
- [GR72] D. D. Givone and R. P. Roasser. Multidimensional linear iterative circuits – General properties. *IEEE Transactions on Computers*, vol. 21, no. 10, pp. 1067-1073, 1972.
- [GYC12] J. Guerra, M. Yagoubi and P. Chevrel. A matrix sign function framework for robust stability analysis and parameter-dependent Lyapunov and Riccati

- equalities. *IEEE Conference on Decision and Control, Maui, Hawaii*, pp. 3490-3495, 2012.
- [GYC13] J. Guerra, M. Yagoubi and P. Chevrel. Parametric Lyapunov equation approach for robust H_2 analysis and structured H_2 control problems. *IFAC System Structure and Control*. Grenoble, vol. 5, pp. 665-670. 2013.
- [GYC14a] J. Guerra, M. Yagoubi and P. Chevrel. A matrix sign function based solution of parameter dependent Sylvester equations. *European Control Conference, Strasbourg, France*, pp. 400-405, 2014.
- [GYC14b] J. Guerra, M. Yagoubi and P. Chevrel. Toward a rational matrix approximation of the parameter-dependent Riccati equation solution. *51th IFAC World Congress, Cape Town, South Africa*, 2014.
- [Ham82] S. J. Hammarling. Numerical solution of the stable non-negative definite Lyapunov equation, *IMA Journal of Numerical Analysis*, vol. 2, pp. 303-323, 1982.
- [HC06] Q. Hu and D. Cheng. The polynomial solution to the Sylvester matrix equation. *Applied Mathematics Letters*, vol. 19, pp. 859-864, 2006.
- [Hig91] N. J. Higham. A collection of test matrices in Matlab. *ACM Transactions on Mathematical Software*, vol. 17, no. 3, pp. 289-305, 1991.
- [Hig97] N. J. Higham. Stable iterations for the matrix square root. *Numerical Algorithms, Springer*, vol. 15, no. 2, pp. 227-242, 1997.
- [Hig08] N. J. Higham. Matrix sign function, functions of matrices: theory and computation, *chapter 5, SIAM edition*, Philadelphia, pp. 107-132, 2008.
- [HMW77a] W. D. Hoskins, D. S. Meek and D. J. Walton. The numerical solution of $A'Q + QA = -C$. *IEEE Transactions on Automatic Control*, vol. AC-22, pp. 881-882, 1977.
- [HMW77b] W. D. Hoskins, D. S. Meek and D. J. Walton. The numerical solution of the matrix equation $XA + AY = F$. *BIT, Numerical Mathematics* vol. 17, pp. 184-190, 1977.
- [How83] J. L. Howland. The sign matrix and the separation of matrix eigenvalues, *Linear Algebra and its Applications*. vol. 49, pp. 221-232, 1983.
- [I-CM-T77] F. Incertis Carro, and J. M. Martinez Torres. An extension on a reformulation of the algebraic equation problem. *IEEE Transactions on Automatic Control*, pp. 128-129, 1977.
- [IM11] B. Iannazzo and B. Meini. Palindromic matrix polynomials, matrix functions and integral representations, *Linear Algebra and its Applications*. vol. 434, pp. 174-184, 2011.
- [IST+04] J. Imae, H. Sagami, T. Kobayashi and G. Zhai. Non linear control design method based on state-dependent Riccati equation (SDRE) via Quasi-Newton method. *43rd IEEE Conference on Decision and Control*, 2004.
- [JL82] J. Jones and C. Lew. Solutions of the Lyapunov matrix equation $BX - XA = C$. *IEEE Transactions on Automatic Control*, vol. AC-27, no. 2, pp. 464-466, 1982.
- [Jon76] E. L. Jones. A reformulation of the algebraic Riccati equation problem. *IEEE Transactions on Automatic Control*, vol. AC-21, pp. 113-114, 1976.

- [KBS94] Ç. K. Koç, B. Bakkaloğlu, and L. S. Shieh. Computation of the matrix sign function using continued fraction expansion. *IEEE Transactions on Automatic Control*, vol. 39, no. 8, pp. 1644-1647, 1994.
- [KL91] C. S. Kenney and A. J. Laub. Rational iterative methods for the matrix sign function, *Journal on Matrix Analysis and Applications, SIAM*, vol. 12, no. 2, pp. 273-291, 1991.
- [KL92] C. S. Kenney, A. J. Laub. On scaling Newton's method for polar decomposition and the matrix sign function. *Journal on Matrix Analysis and Applications, SIAM*, vol. 13, no. 3, pp. 688-706, 1992.
- [KL94] C. S. Kenney and A. J. Laub. A Hyperbolic tangent identity and the geometry of Padé sign function iterations. *Numerical Algorithms* vol. 7, pp. 111-128, 1994.
- [KL95] C. S. Kenney and A. J. Laub. The matrix sign function, *IEEE Transactions on Automatic Control*, vol. 40, pp. 1330-1348, 1995.
- [KLJ89] C. S. Kenney, A. J. Laub, and E. Jonckheere. Positive and negative solutions of dual Riccati equations by matrix sign function iteration. *Systems Control Letters*, vol. 13, pp. 109-116, 1989.
- [KLP92] C. S. Kenney, A. J. Laub, and P. M. Papadopoulos. Matrix-sign algorithms for Riccati equations. *IMA Journal of Mathematical Control and Information*, vol. 9, pp. 331-344, 1992.
- [KLW90] C. S. Kenney, A. J. Laub, and M. Wette. Error bounds for Newton refinement of solutions to algebraic Riccati equations. *Mathematics of Control, Signals and Systems*, vol. 3, pp. 211-224, 1990.
- [KS09] H. Koroğlu and C.W. Scherer. Generalized asymptotic regulation with guaranteed performance: An LMI solution. *Automatica*, vol. 45, no. 3, pp. 823-829, 2009.
- [KS11] H. Koroğlu and C.W. Scherer. Scheduled control for robust attenuation of non-stationary sinusoidal disturbances with measurable frequencies. *Automatica*, vol. 47, no. 3, pp. 504-514, 2011.
- [Lei04] F. Leibfritz. Complib: CONstraint Matrix-optimization Problem library – a collection of test examples for nonlinear semi-definite programs, control system design and related problems. *Technical University of Trier, Department of Mathematics*, URL: www.complib.de, 2004.
- [LBW10] Y. Lin, L. Bao and Y. Wei. Matrix sign function for solving projected generalized continuous-time Sylvester equations. *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2629-2634, 2010.
- [Les02] S. Lesecq. "Equations de Lyapunov et Sylvester", chapitre 5, *Outil d'analyse numérique pour l'automatique*, Paris, Hermès, 2002.
- [LG88] A. J. Laub and J. D. Gardiner. Hypercube implementation of some parallel algorithms in control. *Advanced Computation Concepts and Techniques in Control Engineering*. M. J. Denham and A. J. Laub, editors. Berlin: Springer-Verlag, pp. 361-390, 1988.

- [LP03] V. J. S. Leite and P. L. D. Peres. An improved LMI condition for robust D-stability of uncertain polytopic systems. *IEEE Transactions on Automatic Control* 1, vol. 48, no. 3, pp. 500–504, 2003.
- [Mar84] W. Marszalek. Two dimensional state-space discrete models for hyperbolic partial differential equations. *Applied mathematical modeling*, vol. 8, pp. 11–14, 1984.
- [Mat78] R. L. Mattheys. Stability analysis via the extended matrix sign function, *Proceedings IEE*, vol. 125, no. 3, pp. 241–243, 1978.
- [Mei04] B. Meini. The matrix square root from a new functional perspective: theoretical results and computational issues. *Journal on Matrix Analysis and Applications*, SIAM, vol. 26, no. 2, pp. 362–376, 2004.
- [OP07] R. C. L. F. Oliveira and P. L. D. Peres. Parameter-dependent LMIs in robust analysis: characterization of homogeneous polynomially parameter-dependent solutions via LMI relaxations. *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1334–1340, 2007.
- [PA01] D. Peaucelle and D. Arzelier. Robust performance analysis with LMI-based methods for real parametric uncertainty via parameter-dependent Lyapunov functions. *IEEE Transactions on Automatic Control*, vol. 46, no. 4, pp. 624–630, 2001.
- [Pag99a] F. Paganini. Frequency domain conditions for robust H_2 performance. *IEEE Transactions on Automatic Control*, vol. 44, pp. 38–49, 1999.
- [Pag99b] F. Paganini. Convex methods for robust H_2 analysis of continuous time systems, *IEEE Transactions on Automatic Control*, vol. 44, pp. 239–252, 1999.
- [Par00] P. A. Parrilo. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. *Ph.D dissertation California Institute of Technology, Pasadena, CA*, 135 pages, 2000.
- [Pea62] J.D. Pearson. Approximation methods in optimal control. *Journal of Electronics and Control*, vol. 13, pp. 453–469, 1962.
- [PF00] F. Paganini and E. Feron. Linear matrix inequality methods for robust H_2 analysis: A survey with comparisons, *Advance in Linear Matrix Inequality Methods in Control, Advances in Design and Control*, SIAM, pp. 129–150, 2000.
- [PKL90] P. Pandey, C. Kenney and A.J. Laub. A parallel algorithm for the matrix sign function. *International Journal of High-Speed Computing*, pp. 181–191, 1990.
- [Q-OH98] E. S. Quintana-Orti and V. Hernandez. Parallel solution of Riccati matrix equations with the matrix sign function. *Automatica*, vol. 34, no 2, pp. 151–156, 1998.
- [RL02] M. Rotkowitz, and S. Lall. Decentralized control information structures preserved under feedback, *IEEE Conference on Decision and Control*, 2002.

- [Rob80] J. D. Roberts. Linear model reduction and solution of the algebraic Riccati equation by use of the sign function, *International Journal of Control*, vol. 32, pp. 677-687, 1980.
- [Roe75] R. P. Roesser. A discrete state-space model for linear image processing. *IEEE Transactions on Automatic Control*, vol. AC-20, no. 1, pp. 1-10, 1975.
- [RV09] J. K. Rice and M. Verhaegen. Distributed control of spatially invariant systems in multiple dimensions: a structure preserving computational technique, *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 516-530, 2009.
- [RV10] J. K. Rice and M. Verhaegen. LPV analysis and control using fast iterative solutions to rationally parametric Lyapunov and Riccati equations”, *American Control Conference*, pp. 3877-3882, 2010.
- [Sch06] C. W. Scherer. LMI relaxations in robust control. *European Journal of Control*, vol. 12, no. 1, pp. 3–29, 2006.
- [Sco97] G. Scorletti. Approche unifiée de l'analyse et de la commande des systèmes par formulation LMI. *Thèse de doctorat, Université Paris XI, Orsay*, 282 pages, 1997.
- [SP06] M. Sato and D. Peaucelle. Robust stability/performance analysis for linear time-invariant polynomially parameter-dependent systems using polynomially parameter-dependent Lyapunov functions”, *IEEE Conference on Decision and Control*, pp. 5807-5813, 2006.
- [SR99] D. J. Stilwell, and W. J. Rugh. Interpolation of observer state feedback controllers for gain scheduling. *IEEE Transactions on Automatic Control*, vol. 44, no. 6, pp. 1225-1229, 1999.
- [SSS00] A. Saberi, A. A. Stoorvogel, & P. Sannuti. Control of linear systems with regulation and input constraints. *Communication and Control Engineering Series London: Springer-Verlag, Berlin*, 2000.
- [Sti91] E. U. Stickel. Separating eigenvalues using the matrix sign function. *Linear Algebra and its Applications*, vol. 148, pp. 75-88, 1991.
- [Sto93] A. Stoorvogel. The Robust H₂ control problem: A worst case design, *IEEE Transactions on Automatic Control*, vol. 38, pp. 1358-1370, 1993.
- [TdS01] A. Trofino and C. E. de Souza. Biquadratic stability of uncertain linear systems. *IEEE Transactions on Automatic Control*, vol. 46, no. 8, pp. 1303-1307, 2001.
- [Vou01] P. G. Voulgaris. A convex characterization of classes of problems in control with specific interaction and communication structures, *IEEE American Control Conference, Arlington, USA*, 2001.
- [vWin08] J.-W. van Wingerden. Control of wind turbines with ‘smart’ rotors: Proof of concept & LPV identification. *PhD Thesis, Delft University of Technology*, 160 pages, 2008.
- [YC05] M. Yagoubi, and P. Chevrel. Structured H₂ controller synthesis via a dilated LMI based algorithm. *44th IEEE Conference on Decision and Control and the European Control Conference*, pp. 5746 – 5751, 2005.

- [Yeu84] K. Yeung. Symbolic network function generation via discrete Fourier transform. *IEEE Transactions on Circuits and Systems*, vol. 31, no. 2, pp. 229-231, 1984.
- [YK88] K. Yeung, and F. Kumbi. Symbolic matrix inversion with application to electronic circuits, *IEEE Transactions on Circuits and Systems*, vol. 35, no. 2, pp. 235-238, 1988.
- [YS11] M. Yagoubi and G. Sandou. Particle swarm optimization for the design of static output feedbacks. *18th IFAC World Congress, Milano, Italy*, 2011.
- [ZDG96] K. Zhou, J.C. Doyle, and K. Glover. Robust and optimal control. *Prentice-Hall, Upper Saddle River, NJ*, 1996.
- [ZGB+94] K. Zhou, K. Glover, B. Bodenheimer and J. Doyle. Mixed H_2 and H_∞ performance objectives I: Robust performance analysis, *IEEE Transactions on Automatic Control*, vol. 39, pp. 1564-1574, 1994.
- [Zol77] E. I. Zolotarjov. Application of the elliptic functions to the problems on the functions of the least and most deviation from zero. *Nauk: Zapiskah Rossijskoi Akad, (in Russian)*, 1877.

Thèse de Doctorat

Jérémie Guerra

**Analyse et commande des systèmes paramétrés,
par la fonction signe matricielle**

**Analysis and control of parameter-dependent systems
by use of the matrix sign function**

Résumé

La thèse présentée ici a pour objet de proposer des méthodes de résolution de problèmes d'analyse et de commande de systèmes paramétrés. Ces problèmes sont ramenés à la résolution d'équations de Riccati, de Lyapunov et/ou de Sylvester paramétrées.

S'inscrivant comme une alternative au formalisme LMI nous montrons dans un premier temps que ces équations peuvent être résolues à l'aide de la fonction signe matricielle ou des méthodes d'inversion directe de matrices paramétrées.

Différentes approches sont proposées tout au long du manuscrit. On notera dans un premier temps l'utilisation de méthodes non itératives, se basant soit sur une méthode d'inversion directe, soit sur la définition dite intégrale de la fonction signe matricielle.

Une seconde voie explorée combine la fonction signe matricielle avec la séparation de spectre d'une matrice pour la résolution d'équations de Sylvester non standards.

Ensuite, différentes méthodes, dites itératives, sont proposées également se basant sur des outils mathématiques tels que l'expansion en polynômes de Laurent, les inversions de Fourier, etc.

Tout au long du manuscrit, des exemples sont proposés, tant sur le plan constant que paramétré afin d'estimer l'applicabilité, l'efficacité ainsi que les limites des différentes méthodes proposées.

Mots clés

Fonction signe matricielle, équations de Riccati, Lyapunov et Sylvester, systèmes paramétrés, Polynômes de Laurent, inversion DFT, analyse H_2 robuste et commande H_2 structurée.

Abstract

This thesis focuses on some new methods for analysis and control problems of parameter-dependent systems. These problems are reformulated as a parameter-dependent Riccati, Lyapunov and/or Sylvester equation. As an alternative to the LMI framework, we show initially that these problems can be solved thanks to the matrix sign function, or direct inversion methods of parameter-dependent matrices.

Several approaches are proposed along the manuscript. First the use of non-iterative methods, based either on a direct inversion method, or on the integral definition of the matrix sign function is proposed.

A second way mixes the matrix sign function together with the spectral decomposition of a matrix in an original method for the solution of non-standard Sylvester equations.

Then, several methods, called iterative methods, are also proposed based on different mathematic tools such as the Laurent polynomial expansion, the inverse via Discrete Fourier Transform, etc.

All along the manuscript, examples are shown, simultaneously in the constant and parameter-dependent cases, in order to show to the reader the applicability and the limits of the proposed methods.

Key Words

Matrix sign function, Riccati, Lyapunov and Sylvester equations, parameter-dependent systems, Laurent Polynomials, DFT inversion, Robust H_2 Analysis and structured H_2 control.